

Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros de Telecomunicación



**DESARROLLO DE UNA PROPUESTA DE
ARQUITECTURA PARA EL SOPORTE DE SERVICIOS
HETEROGÉNEOS EN ENTORNOS HOSPITALARIOS**

TRABAJO FIN DE MÁSTER

Augusto Morales Domínguez

2010

Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros de Telecomunicación

**Máster Universitario en
Ingeniería de Redes y Servicios Telemáticos**

TRABAJO FIN DE MÁSTER

**DESARROLLO DE UNA PROPUESTA DE
ARQUITECTURA PARA EL SOPORTE DE SERVICIOS
HETEROGÉNEOS EN ENTORNOS HOSPITALARIOS**

Autor
Augusto Morales Domínguez

Director
Tomás Robles Valladares

Departamento de Ingeniería de Sistemas Telemáticos

2010

Resumen

Actualmente muchos hospitales y otras instituciones de salud están utilizando sistemas de comunicación para optimizar recursos y agregar una nueva generación de servicios a los pacientes. Sin embargo algunos servicios y recursos tecnológicos están limitados a la flexibilidad, disponibilidad y compatibilidad de la infraestructura de comunicación, ya sea por utilización de sistemas cerrados o a las características fijas de una red hospitalaria. Este trabajo de Fin de Master propone el desarrollo de una Arquitectura para el Soporte de Servicios Heterogéneos en Entornos hospitalarios la cual está basada en nuevas tecnologías de Internet, y está enfocada en mejorar la manera en que los servicios son ofrecidos, a la vez que se mantiene la interoperabilidad con estándares médicos de comunicación. Aspectos como la movilidad en casos de emergencias y la interacción con equipo médico son tomados en cuenta con el fin de soportar un futuro despliegue de una plataforma de servicios.

En este documento se propone una Arquitectura de Comunicación enfocada en entornos Médicos que pueda soportar múltiples tecnologías en distintos niveles, tanto de software como de hardware. Además de ello se plantea la adhesión de nuevas tecnologías, que puedan aprovechar al máximo una colaboración entre la red troncal hospitalaria y los distintos nodos y sensores desplegados. Es importante resaltar que para poder soportar una colaboración entre servicios y red, deben existir distintos módulos y middlewares que permitan una abstracción de ciertas capas. En este trabajo se plantea por lo tanto una arquitectura de nodos y servidores genéricos que puedan extender servicios ya existentes, así como también estándares y protocolos como SIP, HL7 y RFID.

Otro de los aspectos que se definen es la inclusión de una plataforma genérica de servicios que permita la inclusión de funcionalidades de manera rápida, flexible, y segura. Para un primer desarrollo se ha elegido el framework OSGi, sobre el cual se implementarán todos los servicios, por su enfoque nativo orientado a la modularidad. Se planteará el uso de tecnologías P4P y P2PSIP en entornos hospitalarios, a la vez que propondrá funcionalidades que puedan dar paso a la inclusión de otro tipo de nuevas tecnologías y desarrollos.

Abstract

Nowadays, hospital and other health institutions are using communications systems in order to optimize resources and add a new generation of available services to patients. However some of these services and technologic resources are limited to flexibility, availability and compatibility of the network infrastructure. Those characteristics are the result of using closed systems and fixed properties in hospital networks. This final Master work is an architecture development proposal for supporting heterogenic services within medical contexts. It is based on Internet's new technologies and focused on enhancing the way services may be offered. Furthermore, it keeps interoperability with communication medical standards. Aspects such as mobility in emergency cases and interaction with medical equipment are taken in to consideration for future services platform development.

In this document I propose a Communication Architecture focused on Medical Environments that can support multiple technologies at software and hardware levels. In addition, the Architecture considers how to add new technologies that can take advantage of the collaboration between core hospital network, different nodes and sensors deployed. In order to support this collaboration between services and network, several modules and middleware are described. These elements will also support abstraction of some network layers. Additionally, some generic nodes and server architectures for extending existing services are depicted with the aim of supporting standards and protocols such as SIP, HL7 and RFID.

One important fact is a service generic platform definition that permits to include several functionalities in a fast, flexible and secure way. In the first prototype, I decided to use OSGi framework, because services can be developed with modularity characteristics. At last I propose to use technologies such as P4P and P2PSIP in hospital environments in order to support future developments.

Índice general

Resumen	i
Abstract.....	iii
Índice general.....	v
Índice de Figuras	ix
Índice de Tablas.....	x
Siglas	xi
1 Introducción.....	1
1.1 Entorno del Trabajo de Investigación.....	2
1.2 Objetivos	4
1.3 Estructura General	6
2 Revisión Tecnológica	7
2.1 Protocolo de Inicio de Sesión: SIP	8
2.1.1 Características y Funcionalidades.....	9
2.1.2 Arquitectura General del Protocolo.....	10
2.2 Open Services Gateway Initiative: OSGi.....	13
2.2.1 Contenedores OSGi.....	15
2.2.2 Especificaciones OSGi.....	17
2.3 HL7: Health Level 7	18
2.3.1 Estructura de Mensajes HL7	19
2.4 Mirth.....	20
2.5 RFID	21
2.6 Sensorización en Entornos Hospitalarios	22
2.6.1 Monitores Biométricos.....	23
2.7 Java Micro Edition (J2ME).....	25
3 Arquitecturas y Servicios para Entornos Hospitalarios.....	27

3.1	Servicios Hospitalarios	27
3.1.1	Clasificación de un Servicio Hospitalario	28
3.2	Casos de Uso	28
3.3	Análisis Tecnológico de Casos de Uso	30
3.3.1	Interoperabilidad con Dispositivos Médicos.....	30
3.3.2	Movilidad en Ambientes Hospitalarios	31
3.3.3	Soporte de Servicios Heterogéneos.....	31
3.3.4	Soporte de Redes Inalámbricas.....	32
3.4	Requisitos de Arquitecturas.....	32
3.4.1	Requisitos funcionales	33
3.4.1.1	Requisitos generales.....	33
3.4.1.2	Requisitos específicos.....	33
3.4.2	Requisitos no funcionales.....	35
3.4.2.1	Rendimiento	35
3.4.2.2	Seguridad.....	35
3.4.2.3	Fiabilidad.....	35
3.4.2.4	Disponibilidad	36
3.4.2.5	Mantenibilidad.....	36
3.4.2.6	Portabilidad de Datos	36
3.4.3	Otros requisitos y Restricciones	36
3.5	Adaptabilidad de Arquitecturas Hospitalarias.....	36
4	Propuesta de Arquitectura General.....	38
4.1.1	Arquitectura de Nodos	40
4.1.2	Arquitectura de Servidor Intermedio.....	44
4.1.3	Arquitectura de Servidor Central	46
4.1.4	Arquitectura y Topología de Red.....	46
4.1.5	Modelo de Datos.....	47
5	Desarrollo de Arquitectura	51
5.1	Consideraciones de Diseño	51
5.2	Arquitectura Detallada de Módulos.....	52
5.2.1	Módulo SIP.....	52

5.2.2	Módulo Mirth.....	53
5.2.3	Módulo de Control.....	54
5.3	Implementación de los Nodos.....	54
5.3.1	Nodos Multifunción.....	54
5.3.2	Nodos de Arquitectura Cerrada.....	54
5.3.3	Implementación en Nodos Móviles.....	56
5.4	Implementación de Servidores.....	58
5.5	APIs Desarrollados.....	59
5.5.1	API SIP J2ME.....	60
5.5.2	API Otras Tecnologías.....	60
5.6	Conclusiones.....	61
6	Arquitecturas de Red Hospitalarias para Situaciones de Emergencia	63
6.1	Arquitecturas P4P.....	63
6.2	Evolución de P4P como Arquitectura para Situaciones de Emergencias.....	64
7	Conclusiones y Trabajos Futuros	68
7.1	Trabajos Futuros.....	69
	Publicaciones Relacionadas con este Trabajo.....	73
	Bibliografía	75

Índice de Figuras

Figura 1. Estructura Cliente/Servidor	11
Figura 2. Diagrama de Elementos.....	12
Figura 3. Arquitectura Software OSGi	13
Figura 4. Contenedores de Java EE Server	16
Figura 5. Ejemplo de Mensaje HL7	19
Figura 6. Funcionamiento de RFID.....	22
Figura 7. Componentes Principales de JavaME.....	26
Figura 8. Arquitectura General Propuesta.....	40
Figura 9. Arquitectura de Nodos	42
Figura 10. Arquitectura Global de Red	47
Figura 11. Características de un iBed	55
Figura 12. Interacción de Arquitectura con Sistemas Cerrados.....	56
Figura 13. Lista de Mensajes Recibidos.....	56
Figura 14. Integración Cliente SIP en Teléfono Móvil	57
Figura 15. Integración de API SIP en Lector RFID	58
Figura 16. Arquitectura de Prototipo de Servidores	58
Figura 17. Bundles OSGi de Servidores	59
Figura 18. Arquitectura P4P para Situaciones de Emergencias.....	66

Índice de Tablas

Tabla 1 – Funcionalidades de Nodos.....	43
Tabla 2 – Funcionalidades de Servidor Intermedio.....	45
Tabla 3 – Funcionalidades de Servidor Central	46
Tabla 4 – Tipo de Tráfico vs Calidad de Servicio	49

Siglas

3DES	<i>Triple Data Encryption Standard</i>
3GPP	<i>3rd Generation Partnership Project</i>
AES	<i>Advanced Encryption Standard</i>
API	<i>Application Programming Interface</i>
B2BUA	<i>Back-To-Back User Agent</i>
EPC	<i>Evolved Packet Core</i>
FTP	<i>File Transfer Protocol</i>
GUI	<i>Graphical User Interface</i>
HIPAA	<i>Health Insurance Portability and Accountability Act</i>
HL7	<i>Health Level Seven</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
IETF	<i>Internet Engineering Task Force</i>
IM	<i>Instant Messaging</i>
IMS	<i>IP Multimedia Subsystem</i>
IP	<i>Internet Protocol</i>
ISO	<i>International Standards Organization</i>
J2ME	<i>Java 2 Mobile Edition</i>
J2SE	<i>Java 2 Standard Edition</i>
JDK	<i>Java Development Kit</i>
MTU	<i>Maximum Transfer Unit</i>
NIST	<i>National Institute of Standards and Technology</i>
OMA	<i>Open Mobile Alliance</i>
ORU	<i>Unsolicited Transmission of an Observation Message</i>
OSGi	<i>Open Services Gateway initiative</i>
OSI	<i>Open Systems Interconnection</i>

P2P	<i>Peer-to-Peer</i>
P2PSIP	<i>Peer-to-Peer Session Initiation Protocol</i>
P4P	<i>Proactive Provider Participation for P2P</i>
PDA	<i>Personal Digital Assistant</i>
PIDF	<i>Presence Information Data Format</i>
PS	<i>Presence Server</i>
PUA	<i>Presence User Agent</i>
RFC	<i>Request for Comments</i>
RTCP	<i>Real Time Control Protocol</i>
RTP	<i>Real Time Transport Protocol</i>
RTSP	<i>Real-Time Streaming Protocol</i>
SDP	<i>Session Description Protocol</i>
SIP	<i>Session Initiation Protocol</i>
SMS	<i>Short Message Service</i>
SQL	<i>Structured Query Language</i>
TCP	<i>Transmission Control Protocol</i>
TIC	<i>Tecnologías de la Información y la Comunicación</i>
TLS	<i>Transport Layer security protocol</i>
UA	<i>User Agent</i>
UAC	<i>User Agent Client</i>
UAS	<i>User Agent Server</i>
UDP	<i>User Datagram Protocol</i>
URI	<i>Universal Resource Identifier</i>
VoIP	<i>Voice Over Internet Protocol</i>
WPAN	<i>Wireless Personal Area Network</i>

1 Introducción

Tradicionalmente hospitales y otros entornos sanitarios son entornos que están en constante cambio y que tienen un comportamiento dinámico. Este comportamiento es causado por múltiples razones: pacientes, personal médico, urgencias y otros factores aleatorios que pueden afectar las tareas y los servicios que han de ser ofrecidos. Adicionalmente estos entornos tienen un sistema de comunicación crítico que a su vez posee requerimientos especiales debido a la información que están destinados a procesar. Además de esto, muchos avances en las arquitecturas de redes, protocolos, servicios, y tecnologías de acceso al medio han sido utilizados en los sistemas de hoy en día. Sin embargo algunas de estas ventajas no han sido aplicadas debido a la falta de iniciativa de adaptación a las restricciones impuestas por los entornos médicos y a los problemas de interoperabilidad entre proveedores de equipos. Los entornos médicos necesitan por lo tanto una arquitectura mucho más específica que cumpla con todos los requisitos tecnológicos y de seguridad que son impuestos, a la vez que pueda abrir la posibilidad de una rápida implementación de nuevas tecnologías.

Los entornos médicos no están solamente limitados a hospitales o a centros de salud, si no que dependiendo del contexto y de la situación deben ser creados de una manera rápida y eficiente, con lo cual el sistema de comunicación debe adaptarse a estos nuevos requisitos impuestos. Está claro también que con el paso de los tiempos ha existido una notable mejora y una rápida adopción tecnológica en los dispositivos médicos, y más allá de esto, se han desarrollado disciplinas como la informática médica y telemedicina que tienen como objetivo acelerar este cambio. Sin embargo, no tendría sentido enfocar todos estos esfuerzos, y no mejorar la manera en que se intercambia la información. Es importante por lo tanto que la evolución de los sistemas de comunicación vaya a la par con las mejoras en dispositivos médicos y sus nuevos requisitos de red.

Existen estudios [1][2] que han pronosticado una evolución de los sistemas telemáticos hospitalarios, y esta realidad se está acentuando cada vez más. Diversas propuestas [3][4] señalan que debe existir una estandarización de los sistemas de información y de las aplicaciones hospitalarias, pero que estas no se deben limitar a aspectos específicos como el formato y la manera en que las bases de datos médicas funcionan. Una estandarización de las redes de sensores, servicios, redes de acceso, y funcionalidades multimedia ayudarían a mejorar tanto el rendimiento general de los

sistemas como una reducción de los costes, aspecto que interesa mucho a entidades encargadas de la salud.

Otro de los factores a tomar en cuenta en el contexto hospitalario son las respuestas de los sistemas de comunicación en situaciones de crisis. Se han propuesto diversos métodos [5] para mejorar la respuesta de estos sistemas, pero aún así, la multiplicidad de requerimientos como: eficiencia, portabilidad, usabilidad, y confiabilidad evitan que se pueda llegar a una metodología, y mucho menos a una arquitectura genérica que pueda servir de apoyo para este tipo de situaciones. Lo que sí es posible es asegurar aspectos como movilidad, gestión de recursos y monitorización utilizando tecnologías ampliamente utilizadas en Internet, pero analizando antes qué características son necesarias, de manera que sean mantenidas sin importar las condiciones del entorno.

Es importante señalar que una arquitectura hospitalaria debe ser capaz de gestionar grandes cantidades de tráfico que puedan generarse en ambientes de emergencias (ya sean atentados, terremotos, guerras etc.). En estos casos los datos transmitidos cobran más importancia que en situaciones cotidianas. Asimismo la fiabilidad del sistema de comunicación será un factor importante en la supervivencia de un paciente. En estos tipos de casos la existencia de mecanismos de gestión de sesiones permitiría la continua provisión de servicios y datos catalogados como críticos. De la misma manera, la adaptación de tecnologías que proporcionan dinamismo a la red supondría un mecanismo para afrontar este tipo de situaciones.

Está claro que las arquitecturas hospitalarias de hoy en día, son una combinación de múltiples tecnologías, protocolos y redes. Estas arquitecturas incluyen comunicación inalámbrica, interoperabilidad de equipos, gestión de alarmas y eventos, etc., por lo que una primera aproximación supondría que no es necesario replantear una nueva arquitectura, con todos los riesgos e inconvenientes que supone.

1.1 Entorno del Trabajo de Investigación

El concepto del Internet de las cosas y la continua demanda de recursos por parte de personas, ya sea pacientes o personal médico, que desean crear, consumir y desplegar servicios son aspectos importantes de cara a la arquitectura de red de un Hospital del Futuro.

Como se mencionó con anterioridad los ambientes hospitalarios son entornos muy dinámicos, y que a su vez tienen requisitos mucho más estrictos si son comparados con otros entornos, como lo podrían ser una red de una empresa, o un centro de procesamiento de datos. De la misma forma, un sistema de comunicación hospitalario al igual que las tecnologías que utiliza, posee características y requerimientos

adicionales que no pueden ser analizados de la misma manera si se compara con otros sistemas informáticos, que utilizan los mismos protocolos. Está claro, que si ocurre alguna falla, las pérdidas podrían ser irreparables, por lo que la total fiabilidad del entorno es un requisito obligatorio. Para hacer un mejor estudio sobre los entornos hospitalarios y sus sistemas de comunicación se pueden definir dos entornos.

- Entorno local de Pacientes: aquí se incluyen todos los equipos o dispositivos conectados al paciente que se encargan de monitorizar su estado y transmitir esta información a la red hospitalaria. Comúnmente existen redes como WPAN que son relacionadas con este entorno. El entorno local de paciente es la fuente principal de los datos generados en un hospital y es la que más importa ya que provee información sobre signos vitales.
- Entorno general Hospitalario: se incluyen todos los sistemas que permiten el funcionamiento, base de datos, equipos de comunicación etc. Este entorno es mucho más generalista y se enfoca en informar al personal médico acerca de la evolución de los pacientes y de todas las actividades dentro del hospital que incluyan a personal médico. Para una clasificación del entorno general hospitalario también se podrían incluir información sobre zonas específicas y sensores que detecten el movimiento de personal a través de ellas.

Según la clasificación mostrada anteriormente una arquitectura de comunicación debe ser capaz de procesar los datos dependiendo del lugar en donde sean generados los eventos. Además de hacia dónde, y hacia quién dirigir la información resultante luego de ejecutarse los procesos existentes.

Realizar un análisis de las necesidades de un servicio desplegado en una red hospitalaria, es un tanto demandante, ya que son múltiples los sistemas que son utilizados para monitorizar pacientes, enviar/recibir alarmas según eventos críticos, lograr una localización efectiva del personal etc. Para ello es necesario primero identificar cuáles son las funcionalidades de primer nivel y cuáles son las opcionales. Finalmente para lograr una interoperabilidad efectiva, debe haber una clasificación de los distintos elementos generadores de información, de manera que puedan soportarse servicios hospitalarios heterogéneos y cooperativos.

Este trabajo identificará las necesidades de la arquitectura de comunicación para que pueda haber una retroalimentación entre distintos entornos, de manera que servicios que sean desplegados en uno o en otro entorno puedan ser consumidos/desplegados recíprocamente. La arquitectura tiene también relación con entornos de AAL (Ambient Assisting Living) ya que aunque no interacciona

directamente con personas que necesiten asistencia, sí provee un mecanismo de mejora a la calidad de servicio de los pacientes y de prestaciones al personal sanitario.

1.2 Objetivos

Dado el actual avance de las Tecnologías de Información y Comunicaciones (TIC) y la creciente popularidad de servicios basados en Internet, se ha visto la oportunidad de llevar múltiples adelantos tecnológicos hacia el plano de los servicios hospitalarios.

El objetivo principal de este Trabajo de Fin de Master es proponer el desarrollo de una arquitectura para el soporte de servicios heterogéneos y que esté enfocada a los entornos hospitalarios.

La arquitectura debe ser compatible con sistemas existentes y ser capaz, en un futuro, de proveer una serie de servicios básicos sobre los cuales se podrían construir otros más complejos. Se tomarán en cuenta los estándares de comunicación utilizados actualmente y las necesidades específicas de adaptación de datos provenientes de sensores.

Se hará una revisión científica sobre las tendencias en arquitecturas hospitalarias y cuáles son las tecnologías que han aportado y que pudieran aportar mayores beneficios a entornos médicos.

Para la consecución de los objetivos de este trabajo se han identificado los siguientes objetivos específicos:

- Dotar a la arquitectura de mecanismos de interoperabilidad. Es necesario facilitar la Interoperabilidad con sistemas hospitalarios existentes ya que es imposible rediseñar, debido a múltiples razones (costos, compatibilidad, capacitación de personal etc.), una arquitectura desde cero. La arquitectura permitirá la interoperabilidad con estándares médicos como HL7, de manera que más que un reemplazo, sea una ampliación de las funcionalidades existentes
- Proporcionar dentro de la arquitectura el soporte a multiplicidad de sensores. En los ambientes médicos son utilizados múltiples sensores, por lo que la arquitectura tomará en cuenta los distintos niveles de importancia, y la relación con las distintas alarmas o eventos que pudieran generarse al llegar a cierto umbral.
- Dotar de mecanismos para la Gestión de Comunicaciones a alto nivel. Dependiendo de la situación unos datos médicos tendrán más prioridad que otros, por lo que la arquitectura deberá ser capaz de gestionar la transmisión

de datos según prioridad. Se debe proveer una negociación efectiva de las sesiones, el ancho de banda, puertos, y otras características que pudieran ser necesarias para mantener la fiabilidad del sistema. La gestión eficiente de recursos también permite que el sistema permita absorber posibles ráfagas de tráfico que puedan ser causadas tanto por malfuncionamiento de elementos de red como por colapso de los canales de comunicación. Además de esto ya que la credibilidad e integridad de los datos es crítica en ambientes de emergencia, se debe asegurar la reservación los recursos necesarios.

- Incluir dentro de la arquitectura características de movilidad y flexibilidad. La movilidad desde el punto de vista hospitalario pudiera no ser catalogada como obligatoria, ya que por la propia estructura fija del hospital no es necesaria. Pero al agregar factores de emergencia, la movilidad toma importancia. La arquitectura proveerá soporte de movilidad tanto a los sensores, como camas y a pacientes, de manera que se añade un grado de flexibilidad a las comunicaciones.
- Adicionar funcionalidades para la gestión efectiva de eventos hospitalarios. Se incorporará un sistema de gestión de alarmas y de presencia, para que sirva de herramienta informativa en caso de que alguna emergencia ocurriese. Las alarmas son elementos importantes dentro de la arquitectura propuesta ya que permiten la interacción de los elementos funcionales con el personal médico, además de que estarán disponibles para cada una de las funcionalidades que puedan ser agregadas en el futuro. El envío/recepción de alarmas se realizará utilizando protocolos estándares de Internet.
- Facilitar la inclusión de elementos dinámicos dentro de la arquitectura de manera que se pueda realizar una gestión del Personal. Se definirá un sistema basado en tecnología RFID, por el cual se tendrá control del tiempo de exposición del personal y de pacientes en áreas catalogadas como delicadas, tales como salas de Rayos X etc. Esta gestión de personal se complementa con el objetivo anterior de gestión de alarmas con el cual se soportará el envío de mensajes de texto, una vez que se ha superado cierto límite de exposición.

La arquitectura de comunicación extenderá las capacidades de los entornos médicos para soportar una nueva generación de servicios médicos, a la vez que se mantiene abierta la posibilidad para integraciones potenciales con sistemas antiguos y redes inalámbricas. El diseño de esta arquitectura tomará en cuenta la inclusión de otro tipo de módulos, como seguridad o gestión de contexto, y que puedan ser incluidos de una manera rápida, sin afectar el funcionamiento de otros módulos. Finalmente se

propondrá una validación de la arquitectura con tecnologías existentes, y con dispositivos reales.

1.3 Estructura General

En este trabajo de Fin de Master, en primer lugar se presenta en el Capítulo 2 una revisión del estado del arte de las tecnologías involucradas en la arquitectura propuesta, haciendo especial énfasis en su aplicación en ambientes hospitalarios.

En el Capítulo 3 se analizan los requerimientos de una arquitectura de comunicación hospitalaria, para luego identificar las funcionalidades que deben ser incluidas. Este capítulo también incluye un análisis de la evolución de una arquitectura hospitalaria que pueda soportar topologías de red dinámicas, y que a su vez permita la provisión de servicios.

En el Capítulo 4 se define la arquitectura de comunicaciones a desarrollar en entornos hospitalarios. Se muestra primero un diseño de alto nivel para luego realizar un diseño detallado de todos los componentes involucrados. Se analizarán además las características necesarias para clasificar los tipos de datos tratados en estos entornos, y los respectivos requisitos que han de ser cumplidos para asegurar la fiabilidad del sistema.

El Capítulo 5 describe el estado del prototipo en desarrollo con el cual se validará la arquitectura propuesta en este documento, y con el cual se pretenden obtener futuros datos que permitan evaluar futuras funcionalidades que deben ser incluidas.

En el Capítulo 6 se analiza la inclusión de tecnologías P4P y cómo las nuevas arquitecturas de composición y despliegue de servicios pueden ser aplicados en ambientes hospitalarios, además de un análisis de los requisitos a nivel de red para un manejo eficiente en situaciones de emergencias, y de alta demanda. Aspectos como la interacción con infraestructuras dinámicas también serán abordados.

Finalmente se presentan las conclusiones y las futuras líneas de investigación por las cuales pudieran extenderse los análisis y conceptos expuestos en este trabajo.

2 Revisión Tecnológica

A lo largo de los años ha existido un amplio desarrollo en las Tecnologías de la Información y la Comunicación (TIC). Este desarrollo ha tenido su origen con las experiencias, exigencias, y requerimientos, de usuarios finales, empresas, operadoras de telecomunicación y el consecuente modelo de explotación comercial que se ha traducido en un beneficiado para todos los actores.

Siguiendo la tendencia actual, el crecimiento tecnológico orientado a los sistemas de médicos ha promovido una rápida implementación y adaptación de recursos con el fin de ayudar a todos los actores que participan en un hospital, que incluyen pacientes, médicos, enfermeras etc. Adicionalmente estos procesos han desencadenado múltiples dudas sobre cómo extender e implementar eficazmente las TIC como herramientas colaborativas a los equipos, procesos y recursos que ya han sido desplegados. En este contexto, múltiples fabricantes han desarrollado sistemas completos que ayudan al despliegue de todos estos elementos. Aunado a esto, la tendencia de estandarización seguida por las TIC ha sido también enfocada en entornos sanitarios, de modo se que plantea una interacción de elementos desplegados por distintos fabricantes.

Uno de los tantos retos sanitarios actuales se enfoca en una pregunta. **¿Cómo las TIC pueden dar lugar a mejores ambientes médicos, que a su vez sean flexibles y que aseguren una interacción entre distintos sistemas?** Al parecer esta pregunta es resuelta con una estandarización de las comunicaciones, datos y servicios, pero en la práctica no es tan tangible. Cada uno de los fabricantes que suplen los entornos sanitarios se enfocan en analizar un entorno (y sus problemas específicos) para luego aportar una solución tecnológica. Estas soluciones cumplen con los estándares dictados por organizaciones como HL7, y HIPAA etc., y contribuyen al enriquecimiento de un entorno hospitalario mediante características a veces novedosas. El inconveniente surge cuando se aspira a que estas características, ya implementadas, interactúen con sistemas abiertos, o con distintos desarrollos que pudieran ser propios de ambientes sanitarios específicos. De la misma manera, estas características podrían ser enriquecidas con funcionalidades provistas por otros fabricantes, sin embargo este enfoque, resulta difícil en la práctica.

Del análisis del párrafo anterior se infiere otra pregunta: **¿Cómo se pueden adaptar tecnologías y soluciones propietarias a una plataforma hospitalaria abierta, de manera que funcionalidades tanto abiertas como propietarias puedan interactuar de forma transparente?**

Para lograr una interacción transparente de diferentes elementos, es necesaria la definición de una arquitectura de servicios genérica que aporte la adición de nuevas funcionalidades, así como su interacción con soluciones previamente existentes dentro de los ambientes hospitalarios. La arquitectura no debe influir en el despliegue de servicios hospitalarios existentes, que se sustenten en sistemas propietarios, ya que forman actualmente el núcleo funcional de muchas arquitecturas de comunicación hospitalarias. Pero a su vez, debe soportar la inclusión de nuevos servicios que ayuden a desplegar un hospital del futuro.

Finalmente, para el diseño de una arquitectura genérica es necesario realizar antes un estudio del estado del arte de tecnologías utilizadas actualmente en ambientes hospitalarios. Este estudio tiene el fin de detectar posibles fortalezas y debilidades, así como sentar las bases de una arquitectura de comunicación flexible y compatible con futuros desarrollos. En este capítulo se proporciona por lo tanto, una visión general de las tecnologías y protocolos relacionados con el ámbito de este trabajo.

2.1 Protocolo de Inicio de Sesión: SIP

El protocolo SIP (Session Initiation Protocol), es un protocolo de señalización de nivel de aplicación definido por la IETF (Internet Engineering Task Force) en Marzo de 1999 con la RFC 2543.

Este protocolo, adoptado por el 3GPP, organismo de estandarización que rige la arquitectura de redes de tercera generación, se ha convertido en un protocolo de comunicaciones fundamental, siendo un elemento permanente en la arquitectura IMS (IP Multimedia Subsystem). Esta tecnología está avanzando a gran velocidad, y cada vez más se utiliza en el desarrollo de sistemas de propósito específico, como es el caso.

Debido al interés y al incesante número de contribuciones para mejorarlo se creó en Septiembre del 1999 un grupo de trabajo para desarrollar el protocolo definiendo métodos, mensajes y los URI's, de los que hablaremos más adelante.

La intención del IETF es que el Protocolo de Inicio de Sesión sea el estándar para el inicio, modificación y finalización de sesiones interactivas de usuario donde intervienen elementos multimedia como el video, voz, mensajería instantánea, juegos online.

SIP soporta movilidad de dispositivos, independencia de la ubicación y tiene unas robustas especificaciones de seguridad. Aunque en un principio el protocolo SIP tenía la finalidad de manejar sesiones de comunicación por Voz sobre IP (VoIP), se ha hecho

muy popular en otras aplicaciones como son el control remoto de dispositivos, la mensajería instantánea, presencia y comunicaciones P2P (Peer-to-Peer SIP).

La importancia de esta tecnología reside en que SIP es un protocolo muy flexible y fácil de extender, definiendo nuevos tipos de mensajes o cabeceras que serán ignorados por los sistemas intermedios que no implementen ese tipo de servicios pero que llegarán intactos a los sistemas finales para su procesamiento.

2.1.1 Características y Funcionalidades

SIP es un protocolo de señalización de la capa de aplicación que permite crear, modificar y terminar sesiones multimedia en una red IP entre dos o más participantes. Es un estándar RFC del IETF (Internet Engineering Task Force), organización internacional responsable de administrar y desarrollar los mecanismos que comprenden Internet. Está especificado en el RFC 3261, que actualiza al ya obsoleto RFC 2543, y se complementa con las distintas extensiones que se han ido desarrollando.

Está inspirado en el protocolo HTTP (HyperText Transfer Protocol), por lo que, al igual que éste, está basado en mensajes textuales intercambiados por los agentes de usuario (UA) y sigue una estructura de petición-respuesta basada en el modelo cliente-servidor. Los agentes de usuario están presentes en los dos extremos de la comunicación y son sistemas inteligentes, pueden actuar como servidores de recursos, identificadores de usuarios, controladores remotos de dispositivos.

SIP proporciona cuatro funciones básicas:

Resolución de direcciones: Mediante el empleo de URI's (Universal Resource Identifier) se puede determinar la dirección física del usuario en cada momento (y también la dirección IP del dispositivo que está utilizando) y comunicarse con él sin necesidad de saber nada más que su identificador URI.

Negociación de una sesión: Los participantes de la sesión pueden determinar qué características utilizar durante la misma, tipo de códec empleado, métodos SIP soportados, protocolo utilizado etc.

Gestión de llamadas: Esta función permite añadir o eliminar participantes en una sesión, informar de redirecciones y también establecer filtros para bloqueo de llamadas, contestador automático, etc. SIP tiene un mecanismo que permite modificar las características de una sesión ya iniciada, volviendo a negociarse con las nuevas características a utilizar.

SIP fue diseñado para funcionar sobre redes IP y generalmente se utiliza junto con otros protocolos como RTP o SDP de los cuales se hablará más adelante. Las sesiones creadas pueden ser de voz, vídeo, chat, juegos interactivos, realidad virtual u otro servicio innovador.

En algunos casos aparecen nuevas aplicaciones en las que tiene sentido trabajar con SIP pero se requiere algún tipo de extensión al protocolo. SIP está preparado para este tipo de situaciones y permite la extensión del protocolo sin interrumpir el comportamiento del núcleo del mismo.

Para definir una extensión se necesita seguir las instrucciones del RFC 4485 Guidelines for Authors of Extensions to the Session Initiation Protocol. Cuando se crea una extensión, se definen nuevos métodos y cabeceras que serán entendidos por los dispositivos preparados para ello e ignorados por aquellos dispositivos que no necesiten ninguna extensión para comunicarse con SIP correctamente.

Una de las extensiones de más importancia para la realización de este proyecto es la que define la funcionalidad de mensajería instantánea (IM). La mensajería instantánea junto con la presencia fueron definidas por el grupo de trabajo IMPP (Instant Messaging and Presence Protocol), aunque no hubo acuerdo sobre el protocolo por el que transmitir los mensajes. Otro grupo de trabajo, SIMPLE (SIP for Instant Messaging and Presence Leveraging Extensions), intentaba definir el modo de implementar la presencia y la mensajería instantánea usando SIP.

SIP/SIMPLE ha sido adoptado en gran cantidad de productos comerciales como el Microsoft Office Live Communication Server y otros de IBM, Nortel, etc. Además SIP/SIMPLE también se utiliza para IMS, por lo que las implementaciones para mensajería instantánea basadas en SIP están ganando aceptación. Además, un sistema de mensajería basado en SIP tiene la peculiaridad de que puede ser combinado con otros servicios de comunicaciones como voz, video, juegos, etc.

2.1.2 Arquitectura General del Protocolo

Las especificaciones de SIP definen una serie de elementos en la arquitectura, que pueden dividirse en componentes finales y componentes intermedios.

Los componentes finales entre los que se establecen las sesiones multimedia se denominan **Agentes de Usuario** (*User Agent* - UA). Estos agentes de usuario estarán en el lado del cliente y en el del servidor. El UAC (*User Agent Client*) es el responsable de la generación de peticiones SIP y de la recepción de las respuestas asociadas.

El UAS (User Agent Server) es el agente responsable de recibir las peticiones SIP y generar las respuestas apropiadas.

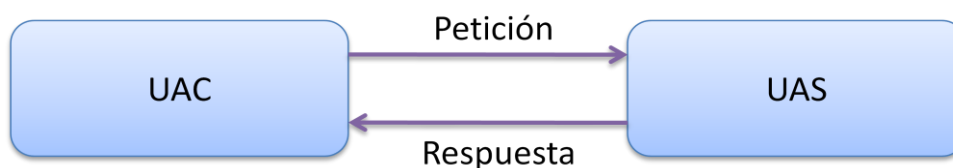


Figura 1. Estructura Cliente/Servidor

Al ser componentes finales, se encuentran en los extremos de la comunicación y generalmente son utilizados por los usuarios finales que interactúan con estos componentes a través de una interfaz gráfica. Además es necesaria una lógica de servicio que mantenga el estado de la llamada y sirva para que todos los componentes se coordinen perfectamente.

Los agentes de usuario SIP pueden implementarse de múltiples formas, por ejemplo pueden residir en un programa software instalado en un PC, como parte de un teléfono fijo o como una aplicación ejecutándose sobre un terminal móvil.

Los elementos intermedios de la arquitectura SIP son denominados servidores SIP, encargados de manejar y direccionar la señalización asociada a múltiples sesiones.

Hay cuatro tipos de servidores SIP:

Servidores Proxy: es un elemento intermedio que realiza peticiones en nombre de los clientes SIP. Su principal tarea es la de encaminamiento, reenviando las peticiones recibidas del agente de usuario o de otro proxy hacia la siguiente ubicación. Hay dos tipos de proxy que merecen nuestra atención, los salientes y los entrantes según si direccionan peticiones salientes o entrantes al dominio en el que están.

Servidores de Redirección: su función es responder a la resolución de nombres y a la ubicación del usuario. Reciben peticiones desde los agentes de usuario o los proxies y responden con información acerca de la dirección del servidor requerido, de tal forma que el cliente pueda contactar con su destino.

Servidores de Registro: También llamados registrar. Es un tipo de servidor que acepta peticiones de registro enviadas por los agentes de usuario. El proceso de registro se explica a continuación: Cuando un servidor “registrar” recibe una petición de registro la analiza para buscar la información de localización del usuario. Si acepta esta petición guardará la información de localización (correspondencia entre la dirección física actual del usuario y el identificador global que se utiliza para localizarlo) en una base

de datos que se llama servidores de localización (Location Service). A partir de este momento el agente de usuario está registrado y puede recibir llamadas multimedia.

Servidores de localización: No es un elemento SIP propiamente dicho, pero como ha sido mencionado anteriormente merece la pena dedicarle un párrafo a esta base de datos. Como hemos dicho en el anterior párrafo, contiene una correspondencia entre direcciones de registro (Addresses of record - AORs) que representan identidades SIP públicas y direcciones de contacto, que representan las dirección reales del usuario en un dominio. Estas dos direcciones están expresadas mediante URIs.

Los servidores SIP pueden operar de dos formas distintas: servidor con estado (stateful) o sin estado (stateless). La diferencia entre ambos modos es que los servidores stateful recuerdan las peticiones entrantes que reciben, junto con las respuestas que devuelven. Un servidor actuando como stateless se olvida de toda la información que recibe una vez que la reenvía.

El último elemento que vamos a nombrar es el llamado B2BUA (Back-To-Back User Agent). El estándar de SIP lo define brevemente como una entidad lógica que recibe peticiones como un UAS y, con el fin de responder a éstas, actúa como un UAC generando peticiones. Es responsable de manejar toda la señalización SIP entre los dos extremos de la llamada desde el establecimiento hasta el final, podemos decir que sirve como “mediador” de la llamada. Este elemento funciona como un proxy, pero impidiendo que un agente de usuario conecte directamente con el UA del destino. Este elemento se usa sobre todo para servidores de aplicaciones para proporcionar funcionalidades avanzadas, como actuar de firewall o modificar las características de la señalización que lo atraviesa.

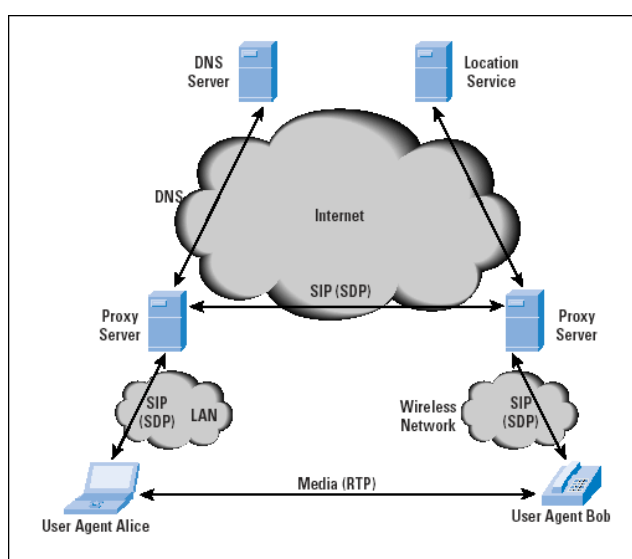


Figura 2. Diagrama de Elementos

2.2 Open Services Gateway Initiative: OSGi

La tecnología OSGi (Open Service Gateway initiative) es un entorno de desarrollo basado en Java y compatible con J2ME (Java 2 Mobile Edition). Este entorno de desarrollo permite manejar componentes dinámicos y las interacciones entre estos y, además, permite a los desarrolladores administrar remotamente el ciclo de vida de las aplicaciones. Por ejemplo, se puede instalar, desinstalar, ejecutar o detener cualquier aplicación desplegada como componente de OSGi.

El “framework” de OSGi fue diseñado inicialmente para ser utilizado en dispositivos con gran porvenir comercial, ya que necesitan ejecutar una gran variedad de software. Éste es el caso de puertas de enlace domésticas (set-top boxes), dispositivos para la automoción y dispositivos PDA (Personal Digital Assistant). Conforme se ha ido descubriendo la potencia de esta tecnología su utilización se ha situado más allá del contexto de puerta de enlace. Por ejemplo, OSGi ha sido seleccionada como la tecnología que constituye el núcleo de la plataforma de desarrollo Eclipse y como la base de aplicaciones empresariales como Java EE Spring.

La plataforma de servicios OSGi fue definida por la asociación internacional OSGi Alliance, fundada en marzo de 1999 y cuyo objetivo es el definir las especificaciones abiertas de software que permitan diseñar plataformas compatibles que puedan proporcionar múltiples servicios.

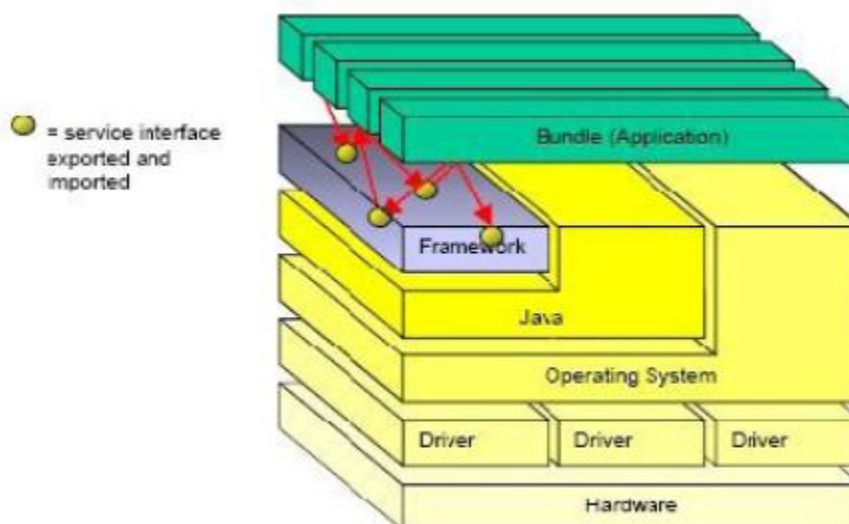


Figura 3. Arquitectura Software OSGi

OSGi define una infraestructura extremadamente eficiente para el desarrollo de aplicaciones basadas en servicios dentro de una máquina virtual Java (JVM).

La parte central de esta infraestructura es el Framework, que implementa un modelo de componentes o aplicaciones dinámico. Estas aplicaciones pueden instalarse, desinstalarse, arrancarse, pararse y actualizarse remotamente, sin necesidad de reiniciar el dispositivo. La plataforma de servicios OSGi es una tecnología que aporta modularidad dinámica a Java y responde a la necesidad de estandarizar la integración de software.

Observando la trayectoria de funcionalidades que incorporan los terminales móviles y, en general, todos los dispositivos tecnológicos podemos afirmar que el futuro está en la convergencia de tecnologías y servicios en un mismo terminal. Los dispositivos móviles actuales realizan una gran variedad de tareas, desde la administración de información personal (agenda, citas, recordatorio de fechas...) hasta el acceso a información en diferentes formatos, pasando por la capacidad de comunicación entre los distintos terminales y otros dispositivos a través de redes heterogéneas.

La integración de todos estos servicios hace necesario que las aplicaciones que se ejecutan en los terminales necesiten estar divididas en módulos. La razón de esta división funcional es muy simple. Los programas “monolíticos”, que incorporan varias funcionalidades sin recurrir a la separación modular, son muy difíciles de desarrollar y de mantener. Además, en aplicaciones profesionales es muy difícil mantener actualizaciones y diferentes versiones cuando el programa no se puede dividir en partes más pequeñas.

Para la administración y el desarrollo de aplicaciones modulares y servicios es necesaria la utilización de una plataforma de servicios como OSGi. Los servicios serán encapsulados en componentes, también llamados “bundles” y el framework de OSGi ofrece la posibilidad de que estos componentes interactúen entre sí, controlen el ciclo de vida de los demás componentes o utilicen sus recursos como propios.

La funcionalidad del Framework está dividida en estos niveles:

Nivel de seguridad: Basada en la seguridad de Java 2 pero añade algunas restricciones y mejora varias características. Interacciona en tiempo de ejecución con el nivel de seguridad de Java.

Nivel de Módulos: Define un modelo de modularidad en Java. Tiene reglas estrictas para el intercambio de paquetes Java entre bundles y también para la ocultación de paquetes a otros bundles que no tienen permiso para usarlos.

Nivel de ciclo de Vida: Proporciona un API de ciclo de vida a los bundles. Este API define como arrancan o se detienen los bundles y como se instalan, actualizan o desinstalan. Requiere que el nivel de módulos esté implementado.

Nivel de Servicio: Proporciona un modelo de programación para los desarrolladores de bundles dinámico, conciso y robusto. Este nivel simplifica el desarrollo y el despliegue de bundles gracias a que separa la interfaz del servicio de su implementación concreta.

Como el framework de OSGi está pensado para integrar una gran variedad de dispositivos con diferente hardware se necesita que el modelo de programación ayude a los desarrolladores en cuestiones de escalabilidad y proporcione una interfaz robusta que permita que componentes heterogéneos se relacionen entre sí.

2.2.1 Contenedores OSGi

Un contenedor se define como un software especializado que permite ejecutar sobre el otro software. Por ejemplo, se puede decir que la máquina virtual de Java (JVM) es un contenedor, ya que monitoriza las aplicaciones Java para administrar la memoria que se está utilizando y para añadir características de seguridad.

OSGi puede definirse como contenedor puesto que permite ejecuciones de programas en su entorno. Además, si se cuenta con las características de portabilidad y movilidad se puede determinar que la tecnología OSGi es idónea como contenedor de aplicaciones móviles.

Los programas comerciales para tecnologías móviles, debido a la complejidad incremental a la que están sometidos (proporcionan características de autenticación, registro, acceso a datos, seguridad en transacciones, etc), necesitan modificar sus características según en el terminal donde se quieran instalar. Esto es como reinventar la rueda. ¿Por qué se tienen que duplicar y modificar las funciones para cada tipo de dispositivo, con el consecuente problema de ineficiencia y de propensión a errores? ¿No tiene más sentido implementar esas características como módulos dentro de contenedores software y permitir al resto de aplicaciones que utilicen estos servicios?

Un contenedor tiene las siguientes características:

Aplicaciones auto-contenidas: Las aplicaciones ejecutadas dentro del contenedor contienen su propio código, librerías y archivos de configuración. La interacción de las aplicaciones y de las librerías puede ser administrada por el contenedor.

Gestión del ciclo de vida: Al llamar a los métodos que gestionan el ciclo de vida de las aplicaciones, definidos en el framework del contenedor, éste puede instalar, comenzar, parar, actualizar y desinstalar cualquier aplicación de forma interactiva o programada.

Servicios estándar: El contenedor proporciona servicios que son comunes a todas las aplicaciones. Por ejemplo un módulo de autenticación puede utilizarse dentro de varios programas y permitirá a las aplicaciones que se autenticquen contra una base de datos.

Servicios hechos a medida: El contenedor debería permitir a sus aplicaciones que ofrezcan sus servicios entre ellas. Esto permitirá reutilizar el código y añadir versatilidad a las aplicaciones modulares.

Un ejemplo de contenedores lo tenemos en la aplicación Java EE Server, su estructura se representa en la siguiente figura:

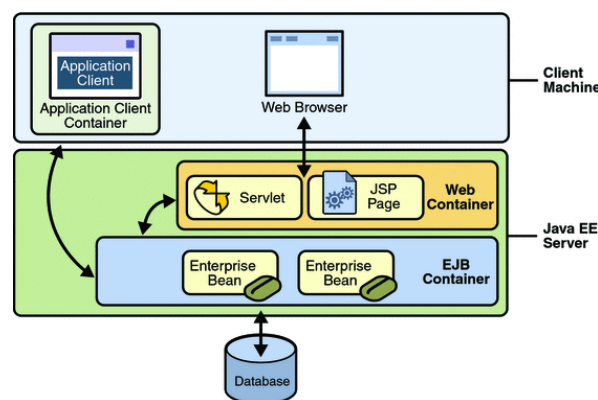


Figura 4. Contenedores de Java EE Server

El servidor Java EE es una porción del producto Java Enterprise Edition y proporciona contenedores EJB y contenedores Web. El contenedor EJB (Enterprise JavaBeans) gestiona la ejecución de servicios basados en Java 2 EE y se ejecutan sobre el servidor Java EE mientras que el contenedor Web permite la ejecución de páginas JSP y de servlets.

También se puede definir un contenedor para aplicaciones del lado del cliente, que gestionará los componentes instalados en él.

La utilización de contenedores proporciona estos beneficios:

Reducción de redundancia en el código: Como existen servicios comunes no es necesario implementar cada uno de esos servicios para cada aplicación y con esto conseguimos reducir la probabilidad de errores en el código.

Actualizaciones automáticas: Cuando los desarrolladores corrigen un error o añaden alguna funcionalidad a algún servicio todas las aplicaciones que usen ese servicio consiguen la actualización automáticamente.

Soporte de modelos de aplicaciones multinivel: Los contenedores tienen la capacidad de que los servicios desplegados ofrezcan separación entre los distintos niveles de aplicación (capas de presentación y comercial).

Movilidad de aplicaciones: Las aplicaciones desplegadas en un contenedor pueden transferirse sin apenas modificaciones a otros contenedores gracias a ser autocontenidas. Esto va acorde con el lema del lenguaje Java “Escríbelo una vez, ejecútalo donde quieras.”

2.2.2 Especificaciones OSGi

La especificación OSGi define una plataforma de servicios (framework) móviles y una serie de servicios contenidos en ésta como APIs de Java para J2ME, J2SE y J2EE.

La versión 2 de OSGi fue liberada en Octubre del 2001 y, gracias a sus muchas implementaciones, ha sido adoptada rápidamente por los vendedores. La última versión liberada ha sido la “Release 3”, en Marzo del 2003.

Las aplicaciones en OSGi están empaquetadas en forma de bundles, que en realidad tienen el formato de un archivo JAR. Como un bundle de OSGi está completamente autocontenido toda la información necesaria para que el bundle funcione debe permanecer dentro del propio bundle. En este caso está contenida en un archivo de manifiesto (MANIFEST.MF).

El contenedor OSGi gestiona completamente el ciclo de vida de los bundles instalados, puede desinstalarlos, actualizarlos, arrancarlos, pararlos y también registrar y desregistrar los servicios que proporcionan esos bundles.

La principal característica de OSGi es que su framework permite gestionar los bundles dinámicamente y sin necesidad de reiniciar el contenedor. Esto hace a la tecnología OSGi muy adecuada para los terminales con aplicaciones móviles.

2.3 HL7: Health Level 7

HL7 (Health level 7) es un conjunto de estándares para el intercambio de información médica. La palabra "Health" (Salud) hace referencia al área de trabajo de la organización y las palabras "Level Seven" (Nivel Siete) hacen referencia al último nivel del modelo de comunicaciones para interconexión de sistemas abiertos (OSI Open Systems Interconnection) de la Organización Internacional para la Estandarización (ISO Internacional Organization for Standarization). HL7 es una organización de desarrollo de estándares (SDO) que funciona en el ámbito de la salud, acreditada por la ANSI (American National Standards Institute - Instituto Nacional Estadounidense de Estándares).

Su especificación más utilizada es un estándar de mensajería para el intercambio electrónico de datos en salud. Proporciona estándares para un dominio particular de la salud, por ejemplo farmacias, imágenes diagnósticas, seguridad del paciente o transacciones de seguros y su dominio comprende información clínica y administrativa. El objetivo final de HL7 es proporcionar estándares de Interfuncionamiento para mejorar la atención sanitaria, optimizar el flujo de trabajo, reducir la ambigüedad y mejorar la transferencia de información entre todos los involucrados.

Dentro de los estándares HL7 se pueden encontrar:

- Mensajería HL7 Versión 2: Estándar de mensajería para el intercambio electrónico de datos de salud.
- Mensajería HL7 Versión 3: Estándar de mensajería para el intercambio electrónico de datos de salud basada en el RIM (Reference Information Model).
- CDA HL7: (Clinical Document Architecture) Estándar de arquitectura de documentos clínicos electrónicos.
- SPL HL7: (Structured Product Labeling) Estándar electrónico de etiquetado de medicamentos.
- HL7 Medical Records: Estándar de administración de Registros Médicos.
- GELLO: Estándar para la expresión de reglas de soporte de decisiones clínicas.
- Arden Syntax: Es estándar sintáctico (if then) para compartir reglas de conocimiento clínico.
- CCOW: Es un estándar framework para compartir contexto entre aplicaciones.

Dentro de los estándares definidos por HL7, se realizará un estudio más profundo sobre las mensajerías versión 2 y 3, ya que son las implementadas dentro del desarrollo que valida la arquitectura propuesta.

2.3.1 Estructura de Mensajes HL7

Los mensajes HL7 son un formato de dato estandarizado que es utilizado a nivel de aplicaciones para entender cuáles son los eventos médicos que están ocurriendo a nivel médico y su relación con pacientes. HL7 define que un mensaje es la unidad transferible más pequeña. Los mensajes HL7 son básicamente eventos generados y enviados cuando ocurre alguna acción en el entorno médico.

La estructura de los mensajes HL7 está compuesta por tres elementos o campos: Message Type, Message event y Message structure.

El HL7 message type es un identificador único para el propósito de un mensaje. Cada mensaje debe contener este campo como una manera de declarar el propósito del mensaje.

El Message Event, el cual también es llamado message trigger, es un identificador del contexto en el cual los mensajes HL7 son generados. El message event consiste en una letra mayúscula seguida de dos dígitos. Por ejemplo A01 es utilizado para un mensaje HL7 de notificación de visita o admisión. Otro ejemplo de mensaje es el A61 es cual representa un evento por el cual se cambia una consulta de un doctor. El tipo de evento es adicionado a la cabecera de cada segmento de mensaje.

El Message structure es la estructura de datos que expresa una asociación entre un message type con un evento, para una clase específica de un mensaje HL7. Cada message structure contiene un identificador único.

```
MSH|^~\&|GHH LAB|ELAB-3|GHH OE|BLDG4|200202150930||ORU^R01|CNTRL-  
3456|P|2.4<cr>  
PID|||555-44-4444||EVERYWOMAN^EVE^E^L|JONES|19620320|F|||153  
FERNWOOD DR.^  
^STATESVILLE^OH^35292|||(206)3345232|(206)752-  
121|||AC55544444||67-A4335^OH^20030520<cr>  
OBR|1|845439^GHH OE|1045813^GHH  
LAB|15545^GLUCOSE|||200202150730|||  
555-55-5555^PRIMARY^PATRICIA P^MD^|||F|||444-44-  
4444^HIPPOCRATES^HOWARD H^MD^<cr>  
OBX|1|SN|1554-5^GLUCOSE^POST 12H  
CFST:MCNC:PT:SER/PLAS:ON||^182|ma/d|70 105|H||F<cr>
```

Figura 5. Ejemplo de Mensaje HL7

Como se mencionó anteriormente la principal ventaja que ha incorporado con la inclusión de HL7 es la estandarización en la manera en que los datos médicos son procesados y guardados, pero existen hoy en día un problema en la implementación en

distintos dispositivos médicos además de un reto[33][34] en la integración con otros sistemas no relacionados con la medicina.

2.4 Mirth

Mirth es un motor (con interfaz HL7) de plataformas cruzadas de código abierto que permite el envío bidireccional de mensajes HL7 entre sistemas y aplicaciones sobre múltiples capas de transporte. Utilizando un bus framework de servicio empresarial y una arquitectura orientada a canales, Mirth permite el filtrado de mensajes, el transformado, y el enrutamiento de los mismos en base a unas reglas definidas por el usuario. Crear interfaces HL7 para los sistemas es fácil utilizando la interfaz web y el asistente para crear canales que asocian las aplicaciones con los componentes del motor Mirth.

Para integrar los servicios con los sistemas HL7 se debe implementar una capa de adaptación para transformar los mensajes entre el dominio de la aplicación y el dominio de HL7. Mirth hace que este paso sea fácil proporcionando el framework para la conexión de sistemas dispares con los protocolos establecidos en los adaptadores y las herramientas de transformación de mensajes.

Mirth utiliza una arquitectura basada en canales para conectar los sistemas con otros sistemas HL7. Los canales consisten en terminales (de entrada y de salida), filtros, y transformadores. Múltiples filtros y una cadena de transformadores se pueden asociar con un canal. La interfaz web de Mirth permite la reutilización de filtros y transformadores en múltiples canales.

Los terminales se utilizan para configurar las conexiones y los detalles de los protocolos. Los terminales de entrada se utilizan para designar el tipo de “listener” para los mensajes de entrada, como por ejemplo TCP/IP o un servicio web. Los terminales de salida se utilizan para designar el destino de los mensajes de salida, como por ejemplo a una aplicación servidora, una cola JMS, o una base de datos.

En cuanto a las características de Mirth, es un software que proporciona una amplia variedad de conectores para escuchar o enviar mensajes HL7 y permite la conexión con diferentes protocolos como pueden ser TCP, MLLP, bases de datos, JMS, SOAP, etc. Mirth es una plataforma cruzada que soporta la mayoría de sistemas operativos y permite la creación y utilización de perfiles de validación para una especificación más eficaz de los destinatarios de los mensajes; por ello aporta versatilidad en cuanto a sistemas operativos y a protocolos. Por otra parte, aunque Mirth es compatible con diversos protocolos de transporte no contribuye con grandes funcionalidades en el

transporte de los mensajes y por ello combinándolo con SIP podrían obtenerse las ventajas de ambos entornos.

Con Mirth todos los mensajes y transacciones se registran en una base de datos interna. Esta base de datos se puede configurar para generar de forma automática respuestas de reconocimiento HL7 (ACK). Mirth está formado por un motor ESB robusto puesto que está basado en el motor Mule ESB para proporcionar velocidad, estabilidad y seguridad en un entorno flexible. Esta es otra de las características importantes en nuestro desarrollo.

2.5 RFID

RFID (siglas de Radio Frequency IDentification, en español Identificación por radiofrecuencia) es un sistema de almacenamiento y recuperación de datos remoto que usa dispositivos denominados etiquetas o tags RFID. Esta tecnología se agrupa dentro de las denominadas Auto ID (Automatic Identification, o Identificación Automática).

El propósito fundamental de la tecnología RFID es transmitir la identidad de un objeto (similar a un número de serie único) mediante ondas de radio.

Una etiqueta RFID es un dispositivo pequeño, similar a una pegatina, que puede ser adherida o incorporada a un producto, animal o persona. Contienen antenas para permitirles recibir y responder a peticiones por radiofrecuencia desde un emisor-receptor RFID.

Las etiquetas pasivas no necesitan alimentación eléctrica interna, mientras que las activas sí lo requieren. Una de las ventajas del uso de radiofrecuencia (en lugar, por ejemplo, de infrarrojos) es que no se requiere visión directa entre emisor y receptor.

El modo de funcionamiento de los sistemas RFID es simple. La etiqueta RFID, que contiene los datos de identificación del objeto al que se encuentra adherido, genera una señal de radiofrecuencia con dichos datos. Esta señal puede ser captada por un lector RFID, el cual se encarga de leer la información y pasársela, en formato digital, a la aplicación específica que utiliza RFID.

Por tanto, un sistema RFID consta de los siguientes tres componentes:

Etiqueta RFID o tag: compuesta por una antena, un transductor de radio y un material encapsulado o chip. El propósito de la antena es permitirle al chip, el cual contiene la información, transmitir la información de identificación de la etiqueta. Existen varios tipos de etiquetas. El chip posee una memoria interna con una capacidad que depende del modelo y varía de una decena a millares de bytes.

Lector de RFID o transceptor: compuesto por una antena, un transceptor y un decodificador. El lector envía periódicamente señales para ver si hay alguna etiqueta en sus inmediaciones. Cuando capta una señal de una etiqueta (la cual contiene la información de identificación de ésta), extrae la información y se la pasa al subsistema de procesamiento de datos.

Subsistema de procesamiento de datos: proporciona los medios para procesar y almacenar los datos.

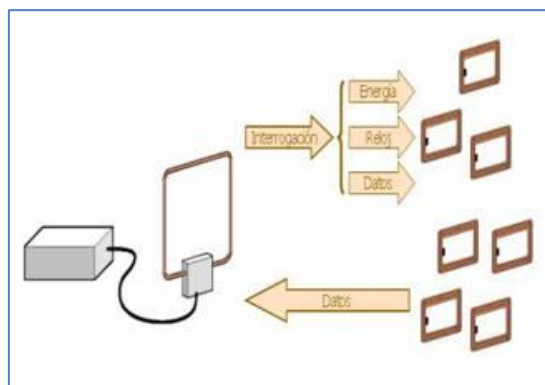


Figura 6. Funcionamiento de RFID

2.6 Sensorización en Entornos Hospitalarios

En la actualidad la sensorización en los entornos Hospitalarios se ha centrado principalmente en tres campos:

Biométrica: Se centra en la recogida de constantes vitales de los pacientes hospitalizados con dos tipos de entornos:

Variables continuas: Variables cuya monitorización es constante y en dónde los criterios de riesgo y estado del paciente pueden variar en cualquier momento.

Variables discretas: Cuya medición se realiza de forma periódica, el valor de la muestra es importante pero se ha de completar con información asociada al entorno o al histórico ya que muchas veces por sí sola no ofrece la información necesaria.

Seguridad: Debido a que nos encontramos en entornos de alta seguridad, existen determinados controles de acceso, riesgos de contaminación, incendios, etc. Para los que se distribuyen por las plantas del Hospital información a través de redes de sensores de propósito específico.

Trazabilidad: Control de activos y personas, debido a la complejidad de las grandes instituciones sanitarias unido al alto coste de los diferentes componentes, cada vez más existe una cierta necesidad de qué sucede en el Hospital en cada momento. También se

busca conocer los procesos en detalle para buscar mejorar en eficiencia de los mismos. En la actualidad y en la práctica, este tipo de tecnologías se centra en la identificación de pacientes y profesionales, el control que se realiza se basa en la interacción de estos con los Sistemas de Información y la vinculación del id con algún proceso concreto como la administración de fármacos, la autenticación en un sistema de información o el paso por un punto que requiera un acceso restringido.

2.6.1 Monitores Biométricos

Sensor es la sonda o elemento final que a través de diferentes mecanismos permite recoger un dato mientras que el monitor es el dispositivo que gestiona la toma y medición de esa muestra. Por ejemplo, en un monitor de temperatura, el sensor únicamente es la sonda que a través del contacto superficial es capaz de medir la temperatura superficial del paciente.

Cuando se habla de Monitores biométricos podemos catalogarlos en base a 4 ejes principalmente:

- Cuándo y cómo se realiza la medida.
- Dónde se realiza la medición.
- Uso principal del sensor
- Tipo de comunicaciones del sensor

Categorización de sensores en base a cuándo y cómo se realiza la medida:

Monitorización continua: Medida que se realiza cuándo el paciente requiere de un seguimiento en un periodo de tiempo extenso, principalmente se aplica en unidades de Cuidados Intensivos, de Reanimación, Quirófanos, ... La medición puede realizarse de forma intrusiva (sensor dentro del propio organismo) o superficial se recogiendo la medida a través del contacto con el paciente. Tiene como objetivo el control exhaustivo de variables vitales en momentos concretos en los que puede peligrar la vida del paciente.

Monitorización discreta: Sensor que permite recoger el valor de una o varias medidas en un momento determinado del tiempo. Se realiza cuándo ha sido pautado o prescrito por un profesional competente. Aquí se incluyen la mayoría de medidas que se realizan cuándo un paciente está ingresado en planta hospitalaria. Ofrece la visión de una variable concreta en un momento concreto.

Otra forma importante de categorizar las variables es identificando dónde se realiza la medición:

- **Gabinete:** Son aquellas medidas que se realizan siempre en un espacio concreto o bien porque el sensor o monitor, es de gran tamaño o coste y requiere trasladar al paciente al lugar (p.ej. monitores o dispositivos de imagen diagnóstica) o bien porque se realizan en un espacio específico en dónde se encuentra el especialista y los medios asociados a esa medición (p.ej. prueba de esfuerzo (cardiología), espirometría (tratamiento respiratorio))
- **Hospitalización:** Aquellas medidas que se realizan en la planta de Hospitalización mientras el paciente está ingresado. Generalmente suelen tratarse de medidas discretas.
- **Salas de críticos:** Espacios como las unidades de reanimación, áreas prequirúrgicas, unidades de cuidados intensivos, determinados boxes de urgencias. En todos estos espacios, es necesario que el paciente se encuentre monitorizado de una forma especial ya que se encuentra en un estado crítico.

Cuando se habla de uso principal del Sensor se puede categorizar en:

- **Diagnóstico:** La finalidad de la medición es poder diagnosticar o identificar la patología asociada a un paciente. La suele practicar un médico/ técnico especialista.
- **Control:** La medida tiene como objetivo controlar al paciente, para que una determinada variable genere una alerta en caso de cruzar un determinado umbral fijado por el médico.
- **Rehabilitación/ Tratamiento/ Ejercicio paliativo:** son aquellos monitores cuya funcionalidad adicional a la recogida de una determinada muestra, tiene como objetivo ayudar al tratamiento de la persona, en este ámbito pueden entrar desde bombas de insulina, a dispositivos de electroestimulación.

La última categorización, identifica como se puede interactuar con el monitor biométrico.

Sin comunicaciones: En este apartado se agrupan monitores de más de 10 años, y en algunos fabricantes de hasta 5 años. Son sensores que cumplen con su funcionalidad clínica pero que no facilitan la recogida de de datos desde otro equipo o el envío de la información a otro equipamiento. Suele incorporar un periférico para mostrar el resultado que facilita o bien la impresión en papel o la visualización a través de una pantalla.

Comunicaciones cable: La gran mayoría de los sensores, tanto los de bajo coste (p.ej. glucómetro) como sensores más complejos ya integran comunicaciones tipo cable como serie RS232, USB, Ethernet etc.

Comunicaciones inalámbricas: Suelen tratarse o bien monitores de última generación o de gama alta. Las comunicaciones suelen ser:

- Infrarrojos: En este caso son sensores antiguos de más 5 de años de fabricación que facilitaban un nuevo mecanismo de acceso desde el exterior.
- Bluetooth: aquellos orientados a consumo e integración con plataformas de telemonitorización a domicilio, telemedicina, etc.
- Zigbee: algunos fabricantes comienzan a desarrollar primeros prototipos de monitores con este tipo de comunicaciones. Son primeras unidades y no están accesibles en grandes unidades
- WIFI: los principales fabricantes de dispositivos Phillips, General Electric, etc disponen de gamas altas de producto que incluyen comunicaciones WIFI. Especialmente orientadas para monitorización continua de pacientes críticos, incluyen en muchos casos hasta envío de datos sobre estándares.

2.7 Java Micro Edition (J2ME)

JME es un subconjunto de la plataforma Java, de Sun Microsystems. Provee de los API's necesarios para la programación de pequeños dispositivos de consumo como pueden ser PDA's, teléfonos móviles, etc.

Es una opción muy buena para la creación de juegos o pequeñas aplicaciones para teléfonos móviles, además de que incluye muchas APIs orientadas a conectividad.

Una de las características más importantes es que al utilizar tecnologías Java, las aplicaciones desarrolladas se pueden portar en una gran cantidad de dispositivos distintos. Aunque se presentan numerosos problemas de portabilidad, lo que implica realizar al menos pequeñas adaptaciones de una misma aplicación para distintas familias de terminales.

JME no es propiamente un sistema operativo, sino una plataforma de desarrollo, y la máquina virtual (VM) que ejecuta las aplicaciones puede estar implementada sobre múltiples sistemas operativos. Sin embargo, en numerosos terminales de gama media/baja, que utilizan sistemas operativos propietarios y cerrados es la única plataforma que permite añadir aplicaciones de terceros. Por otro lado, se trata con

diferencia de la plataforma de desarrollo más extendida en el parque de terminales móviles.

Debido a la inmensa variedad de dispositivos móviles que soportan la tecnología Java ME, existen actualmente diferentes implementaciones para cada tipo de dispositivo. Específicamente cada dispositivo está basado en tres componentes principales.

- Configuration: Define las librerías básicas y la versión de la Máquina Virtual Java (VM).
- Profile: Define una serie de API (Application Program Interface) que están disponibles para el dispositivo.
- Package: define una serie de APIs opcionales que pertenecen a una tecnología específica. Por ejemplo SIP, Bluetooth, LWUIT etc.

En el documento específico se harán uso posteriormente de los APIs SIP y bluetooth para lograr un desarrollo de un prototipo de nodo en terminales móviles.

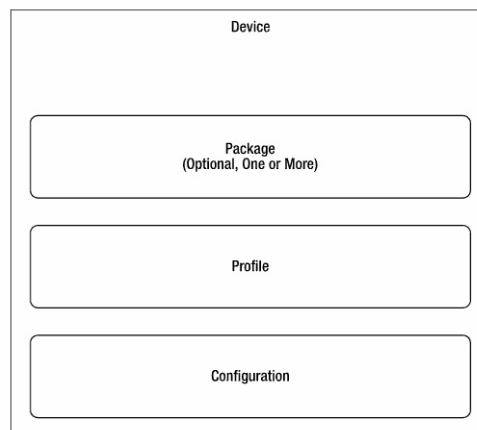


Figura 7. Componentes Principales de JavaME

3 Arquitecturas y Servicios para Entornos Hospitalarios

Una arquitectura de comunicación es la definición de todos los protocolos, redes, datos, nodos, y tecnologías involucradas en la provisión de un servicio específico. Por lo tanto, es importante demarcar cuál es el alcance de la arquitectura de comunicación en relación a los servicios que serán ofrecidos y sus requerimientos, los cuales tendrán una relación directa con el tipo de red que será desplegada.

Las arquitecturas de comunicación abordan los problemas de una manera global, a la vez que ayudan a planificar, evaluar y seleccionar los distintos componentes necesarios para que un sistema funcione correctamente, además de las consecuentes necesidades de interoperabilidad que puedan surgir en la implementación.

Una arquitectura de comunicación hospitalaria, es necesaria, ya que describe las características que debe tener una plataforma de servicios hospitalarios. Esta plataforma tiene a su vez requisitos que deberán ser cumplidos en la propia implementación, y que estarán sujetos al diseño funcional de la arquitectura original. El correcto diseño de una arquitectura tiene relación directa con las futuras mejoras con respecto a servicios y funcionalidades en múltiples ambientes hospitalarios, además de la interoperabilidad con dispositivos que ya estén desplegados.

En este capítulo se analizarán qué tipos de servicios hospitalarios se desean ser soportados por la arquitectura, los casos de uso que los representan y los requisitos impuestos. Finalmente se analizarán aspectos como la adaptabilidad y compatibilidad de la arquitectura propuesta, así como los dispositivos lógicos/físicos, servicios y las interfaces por las cuales interactúan.

3.1 Servicios Hospitalarios

Un servicio es, según la terminología SOA, *una entidad de computación independiente que forma parte de una arquitectura distribuida y que puede establecer enlaces de interacción con otros servicios de forma dinámica*. Un servicio hospitalario no es más que una extensión de esta definición en la cual los datos provienen de un contexto médico específico. Estos tipos de servicios tienen requisitos especiales que serán analizados posteriormente y que afectan la manera en que la arquitectura de red es planteada. Además de ello se deben definir elementos como consumidores de servicio, creadores de servicios, prioridad de datos y la jerarquía topológica que soportará la red.

3.1.1 Clasificación de un Servicio Hospitalario

Según el enfoque de servicios heterogéneos propuesto en la arquitectura se realiza una clasificación según los tipos de servicios que serán soportados.

- Servicios orientados a la plataforma de comunicación: estos servicios incluyen todas las funcionalidades básicas que deberá tener la arquitectura. Este tipo de servicios además permitirá asegurar la robustez y dinamismo del sistema. Entre estos tipos de servicio se pueden incluir, monitorización de dispositivos, captura de datos de sensores, calidad de servicio según prioridades etc. Además de ello estos servicios podrían apoyar a ciertos elementos de red exclusivamente dedicados a funciones médicas como lo podrían ser tomógrafos, ultrasonidos etc.
- Servicios orientados a personal médico: estos servicios tienen la función de servir de nexo entre la plataforma de comunicación y el ambiente en donde se desenvuelve el personal. Además, proporcionan una interacción y control de las funcionalidades básicas de la red y los dispositivos médicos. Ejemplo de estos servicios son: mensajería, alertas, interfaces de usuario con dispositivos etc. Esta clase de servicios podrían apoyarse en los servicios básicos proveídos por la plataforma, así como facilitar una interacción con personas.
- Servicios orientados a pacientes: estos servicios tienen el objetivo de ofrecerle al paciente una mejor experiencia, y por lo tanto se basan en servicios muy habituales. Ejemplo, televisión, radio etc.

Para realizar una aproximación de una arquitectura hospitalaria que soporte las funcionalidades y los distintos tipos de servicios descritos anteriormente es necesario primero, detectar casos de usos específicos en los cuáles se note una clara necesidad de nuevos elementos. Estos elementos pueden estar incluidos ya dentro de entorno hospitalario, pero no dentro de una plataforma centralizada que permita su reutilización, por lo que posteriormente se describirán los requerimientos específicos a cumplir por esta plataforma.

3.2 Casos de Uso

En un escenario ilustrativo, una emergencia ya sea un terremoto, un atentado terrorista o cualquier desastre de cualquier otra índole ha ocurrido. Un paciente el cual se encuentra en una cama temporal de un hospital, ya que para ese momento no existen camas fijas disponibles, se le han asignado una serie de sensores de constantes vitales. Estos sensores son de carácter móvil y forman parte de una categoría diferentes de sensores que comúnmente son utilizados en estos casos. Está claro que

los datos provenientes de ellos deben ser enviados a un sistema alternativo para que médicos o enfermeras tengan la capacidad de analizarlos. Este sistema alternativo pudiera encontrarse físicamente cerca o lejos del paciente, por lo que en el segundo caso, debe existir una infraestructura de comunicación, que en este caso específico debería ser inalámbrica. En este punto se añaden tanto al sensor como a la infraestructura que lo soporta factores como fiabilidad, movilidad y seguridad. Continuando el caso anterior, si las constantes vitales alcanzan un límite crítico, el paciente deberá ser atendido inmediatamente, y en tal caso debe informarse tanto a médicos como enfermeras del estado crítico de este.

El escenario anterior tiene diversos requisitos, como lo son la movilidad ya que el paciente está utilizando una cama temporal, y no hubo tiempo de extender la infraestructura de comunicación hacia él. Un Segundo requisito sería el manejo eficiente de recursos. Ya que existe una situación de emergencia, si al mismo caso de uso se le añade una falla en las comunicaciones primarias, debe haber una infraestructura de apoyo que permita gestionar de manera eficiente la prioridad de los distintos datos médicos que son transmitidos a través de la red, tomando en cuenta que esta última no tiene la capacidad, ancho de banda, y recursos de la primaria. En este caso por lo tanto se debe asegurar que las comunicaciones críticas provenientes de sensores sean llevadas a su destinatario, así como las alertas que sean enviadas al personal médico. Por lo tanto el sistema de apoyo deberá contar con una categorización de datos que sea flexible a la situación que esté ocurriendo en ese momento.

En otro escenario mucho más específico, un paciente se acerca a un centro de atención sanitaria con una fuerte dolencia. El primero paso será un análisis de su estado de salud, por parte del personal médico. Luego de que se detecte que es necesario ingresarlo se le asignará una cama. Esta cama tendrá un dispositivo que será el concentrador de todos los sensores médicos, por ejemplo un iBed. Tanto los sensores como el iBed utilizarán un tipo de mensajería basadas en estándares HL7 para transmitir toda la información concerniente a las constantes vitales. Adicionalmente el paciente necesitará moverse alrededor de ciertas áreas catalogadas como de alta peligrosidad, por ejemplo: cuarto de rayos X, laboratorios, cuartos con láseres etc., a la vez que la información tomada debe ser transmitida al puesto de enfermería para la verificación continua de las constantes vitales. Otro aspecto importante es cuando el tratamiento médico es aplicado al paciente; este tipo de medicinas será tomada y administrada manualmente por el personal médico, por lo que al final todo proceso médico recaerá al final en el factor humano.

Según el escenario anterior se pueden definir algunas mejoras en los procesos hospitalarios. Por ejemplo la cantidad de medicación administrada podría ser rastreada, con lo que la trazabilidad se podría mantener desde la farmacia que la supe, hasta los mismos pacientes. Esta mejora tendría especial importancia en medicinas costosas o en tratamientos aplicados a enfermedades catalogadas como de alta peligrosidad. Otra mejora sería la existencia de algún sistema que le informe al personal médico acerca del tiempo de exposición en lugares peligrosos, ya sea utilizando RFID tags, muy extendidos actualmente[18], u otras tecnologías. Los datos capturados podrían ser mostrados al personal médico en una manera amigable, y a su vez extendiendo la posibilidad de que ellos puedan configurar sus alertas, o procesador esta información en algún dispositivo móvil, tipo iPad. Esta última mejora incluiría también una re-planificación de actividades del personal ya que también se podría tener acceso continuo a información sobre salas o equipos disponibles en el hospital.

En un tercer escenario más tecnológicamente orientado, sensores concentradores tipo iBed, o dispositivos móviles para personal médico[19] son desplegados en el hospital. Estos tienen su propia interfaz de red, y podrían incorporarse dinámicamente a la red hospitalaria, con lo que aumenta la interacción con los sistemas y servicios hospitalarios existentes. Este escenario, incorporaría múltiples tecnologías de acceso al medio, como lo podrían ser Wi-Fi, Zigbee o Bluetooth, pero utilizando una pila de protocolos basadas en TCP/IP, por lo que se podrían aprovechar la uniformidad a nivel de transporte y aplicación, y lograr una interoperabilidad entre distintos dispositivos. Además de esto la inclusión de redes basadas en P4P añadirían un soporte colaborativo entre una red hospitalaria que se acaba de crear, y otra red fija que se encuentra operativa en ese momento. Esta red se encargaría de suplir funcionalidades que no puedan ser soportadas por la red dinámica recién creada y que tengan que ver con procesamiento, almacenamiento o interacción con otros sistemas no hospitalarios.

3.3 Análisis Tecnológico de Casos de Uso

3.3.1 Interoperabilidad con Dispositivos Médicos

Existen tecnologías esenciales que han sido mencionadas en los casos de uso anteriores. Como se había mencionado anteriormente HL7 ha propuesto un número de estándares que garantizan la comunicación e interoperabilidad entre distintos sistemas informáticos utilizados en hospitales y otros entornos sanitarios. Como resultado de este estándar, existen distintas implementaciones de código abierto que tienen el fin de integrar un motor para el procesamiento de este tipo de mensajes, uno

de estos es Mirth. Mirth, como se introdujo en el capítulo anterior, está basado en una serie de librerías que permiten analizar efectivamente mensajes basados en HL7, a la vez que provee un framework completo para la interacción con distintos dispositivos médicos. Adicionalmente a este existen otras librerías como HAPI[6], basada en Java, que tienen el mismo fin.

Existen por lo tanto diversas desarrollos orientados a agregar soporte HL7. Sin embargo existen puntos en contra detectables en estas iniciativas. Entre ellos se encuentran:

- Falta de soporte total al estándar.
- Dependencia en protocolos inferiores en el caso de Mirth.
- Problemas de interacción con dispositivos empotrados.
- Implementación en único lenguaje de programación (HAPI Java)

Como consecuencia, estas deficiencias se vuelven factores importantes en ambientes en donde se necesita un despliegue rápido y flexible de servicios médicos.

3.3.2 Movilidad en Ambientes Hospitalarios

Como se mencionó en el capítulo 2, SIP es un protocolo de señalización utilizado para crear, modificar y transmitir mensajes sobre internet. Actualmente existen estudios [7][8] que sugieren la integración de SIP en entornos médicos. Además de esto SIP es el protocolo de referencia en el núcleo de la red de siguiente generación (NGN) y del IP Multimedia Subsystem (IMS). SIP ofrece una manera eficiente para agregar características de movilidad, calidad de servicio, y manejo de sesiones, sin afectar la compatibilidad con otro tipo de protocolos. Esta particularidad permite mantener la división entre señalización y datos. Otra característica importante de SIP es la flexibilidad en la implementación y la capacidad de asegurar fiabilidad entre distintos nodos con la utilización de Proxys.

En los casos de uso específicos, se detectan dos ventajas importantes que aporta SIP: el soporte a las comunicaciones móviles y la integración con redes de sensores. Finalmente en este contexto, se establece que la utilización de SIP es válida, como protocolo base de comunicaciones en arquitecturas hospitalarias.

3.3.3 Soporte de Servicios Heterogéneos

Los casos de uso anteriores exigen que existan características dinámicas y que haya una interacción entre distintos elementos con distintas tareas. Esto se debe a que los datos generados por un elemento o servicio deberán estar disponibles y ser

consumibles para otros, de manera que se creen, a través de múltiples piezas de software, servicios dinámicos.

Otra tecnología que ha demostrado una alta flexibilidad para adaptarse a entornos médicos es OSGi. Como se mencionó anteriormente su concepto principal se basa en proveer un ambiente estandarizado para pequeñas aplicaciones llamadas “bundles” y que permitan interactuar entre ellas. Por estas características la utilización de OSGi [9][10] ha probado ser una solución válida para el despliegue de arquitecturas de servicios en ambientes hospitalarios. Ya que OSGi realiza la interconexión entre diferentes piezas de software, sensores, alertas, y otros tipos de datos pueden ser integrados en un sistema sanitario.

3.3.4 Soporte de Redes Inalámbricas

Existen distintas condiciones en las cuales una tecnología inalámbrica puede ser útil en el despliegue de servicios médicos. Estas condiciones están ligadas a las propias características de la red, la tecnología que se esté utilizando, las exigencias de fiabilidad, no interferencia etc. Por ejemplo en un ambiente hospitalario, es crítico contar un sistema de respaldo en algunos lugares específicos, como lo podrían ser las Unidades de Vigilancia Intensiva (UVI). En estos casos las redes inalámbricas pueden convertirse en una alternativa útil ya que no requieren que los equipos estén cableados para poder operar. Como resultado de esto, están emergiendo tecnologías como Wireless Body Area Network (WBAN)[11] que intentan solucionar algunos problemas relativos a las redes de sensores y su interacción con pacientes. Otras soluciones [12][13] proponen arquitecturas de red inalámbricas para comunicar distintos dispositivos médicos y soportar futuros servicios médicos. Por lo tanto es comprensible que el soporte a redes inalámbricas es una característica que puede ser tomada en cuenta para extender la gestión de los sistemas de comunicación hospitalarios, y el soporte nativo a la movilidad.

3.4 Requisitos de Arquitecturas

En función de los casos de uso descritos anteriormente se pueden identificar nuevos requisitos, los casos de uso identificados son: Verificar la administración de un fármaco o sangre, monitorizar la trazabilidad de los medicamentos, trazabilidad de pacientes, personal médico, monitorización de las constantes del paciente, monitorización del estado de las consultas, administración de medicamentos y otros procedimientos, control del tiempo de exposición del personal y soporte a movilidad de sensores.

3.4.1 Requisitos funcionales

Los requisitos funcionales, son características que debe tener la arquitectura hospitalaria propuesta, con el fin de cumplir tareas específicas. En este caso, tanto las tecnologías como la propia forma en que está diseñada el sistema derivado de la arquitectura, deben asegurar que se cumplan los requisitos.

3.4.1.1 Requisitos generales

- **Requisito 1:** El sistema deberá enviar avisos a los médicos con la llegada de cada nuevo paciente al hospital.
- **Requisito 2:** El sistema deberá enviar avisos a los celadores con cada alta de pacientes que se produzca, para acomodar la habitación para un nuevo paciente.
- **Requisito 3:** El sistema deberá avisar mediante mensajes al médico cuando se detecte unos valores anómalos en alguna de las constantes monitorizadas del paciente.
- **Requisito 4:** El sistema enviará un mensaje a enfermería para avisar que no se ha administrado un medicamento.
- **Requisito 5:** El sistema enviará un mensaje al responsable del área avisando de sobreexposición del personal a agentes contaminantes.
- **Requisito 6:** El sistema enviará una alarma al responsable del área cuando el personal no ha cumplido con los tiempos de descanso tras la exposición a agentes contaminantes.
- **Requisito 7:** El sistema deberá ser capaz de comunicarse con los elementos o dispositivos que monitorizan las constantes de los pacientes, de manera que los datos estén disponibles para cualquier elemento o parte de software que así lo requiera.
- **Requisito 8:** El sistema deberá contar con dispositivos inalámbricos compatibles para la recepción de las alertas. Las alertas generadas podrían ser de tipo multimedia (audio, video) o utilizar características propias del dispositivo (vibración, luces etc.)

3.4.1.2 Requisitos específicos

- **Requisito 9:** La arquitectura de comunicaciones debe ser capaz de intercambiar los mensajes entre los diferentes nodos. Estos mensajes deberán ser adaptados según el tipo de datos que desea transmitirse por lo que debe haber antes una negociación entre nodos.

- **Requisito 10:** El sistema deberá adecuarse a la arquitectura existente en el hospital. La arquitectura propuesta no debe influir, ni degradar las funcionalidades ya existentes, sino que debe haber cierta adaptabilidad en el diseño.
- **Requisito 11:** El sistema deberá ser capaz de proporcionar comunicación entre los dispositivos móviles mediante una infraestructura de red inalámbrica. Esta red inalámbrica y su correspondiente característica de movilidad deberá ser tomada en cuenta para que haya siempre una localización lógica de los nodos.
- **Requisito 12:** El sistema deberá ser capaz de encaminar los mensajes diferenciándolos según la importancia de los datos transmitidos. Ya que pueden existir múltiples datos, la arquitectura debe ser capaz de diferenciar estos datos y asegurar una mínima calidad de servicio, con el uso de distintos protocolos.
- **Requisito 13:** El sistema en caso de caída de un servidor deberá ser capaz de distribuir todas las comunicaciones a otro servidor para no perder información. Esta redistribución debe ser realizada automáticamente, de manera de que las capas de software superiores no sean conscientes que hubo un problema, al menos que se les mencione específicamente.
- **Requisito 14:** El sistema en caso de caída de la red deberá cambiar a una red de comunicaciones alternativa inalámbrica. Los distintos dispositivos de red deben contener funcionalidades que les permitan conocer el estado de la red, de modo que exista una red de respaldo para las comunicaciones.
- **Requisito 15:** El sistema debe intercambiar los datos médicos mediante estándares HL7 para proporcionar una mayor interoperabilidad al entorno hospitalario. Ya que HL7 es uno de los mayores estándares utilizados, se debe asegurar la interacción con distintos dispositivos médicos que lo usen. Debe existir también la posibilidad de una recodificación de estos mensajes en otros formatos de datos para una posterior reutilización e interacción.
- **Requisito 16:** El sistema deberá permitir definir las prioridades de los datos que se envían permitiendo así adaptarse mejor a las diversas situaciones de los pacientes. Dependiendo del contexto de salud del paciente pueden haber situaciones en que la prioridad de los datos capturados por ciertos sensores sea dinámica, por lo que la arquitectura debe soportar este modelo flexible.
- **Requisito 17:** El sistema deberá gestionar los equipos dependiendo de su estado y jerarquía en la arquitectura. Ya que la capacidad de distintos nodos es variable, la arquitectura debe contemplar la comunicación y procesamiento de datos jerárquico.
- **Requisito 18.** El sistema deberá almacenar y distribuir la información médica y los historiales de forma segura y con protección frente a posibles pérdidas de datos en la red. Debe existir una separación lógica que permita que aunque distintos datos

sean gestionados por un mismo nodo, sólo sea permitido el acceso por elementos de software diseñados como para ello.

- **Requisito 19:** la arquitectura debe contemplar la reutilización de funciones que ya estén implementadas dentro del sistema. Tanto los datos como los elementos de la arquitectura deben poder interactuar bajo la orquestación de un elemento o nodo principal. Debe poder soportarse la futura inclusión de módulos y de distintos tipos de topologías de red que permitan expandir la utilizada del sistema.

3.4.2 Requisitos no funcionales

3.4.2.1 Rendimiento

El rendimiento del sistema depende en gran medida de las dimensiones del hospital puesto que el número de pacientes va a condicionar el volumen de información intercambiada. La arquitectura de comunicaciones está formada por un nodo o elemento central encargado de controlar los nodos intermedios. Los nodos intermedios dividirán el volumen de procesamiento, ya que cada servidor se encargará de un área hospitalaria específica. En cada una de estas áreas habrá un número variable de pacientes. Para asegurar el rendimiento del sistema también habrá que tomar en cuenta el número de nodos móviles que serán fuente o destinatarios de datos.

3.4.2.2 Seguridad

La seguridad en las comunicaciones hospitalarias es de gran importancia no sólo por el ambiente crítico en donde se desarrollan sino también por la sensibilidad de los datos transmitidos. Los mensajes de información médica en formato HL7 deberán ir cifrados con algún tipo de algoritmo como lo puede ser, AES, 3DES etc. Todos los accesos a la información almacenada requerirán de autenticación para evitar acceso a información no autorizada o modificación de datos.

3.4.2.3 Fiabilidad

Como se mencionó con anterioridad, puesto que la información del hospital es crítica, la fiabilidad del sistema de comunicación es una característica obligatoria. El sistema tiene que ser capaz de informar de situaciones inmediatamente después de ser detectadas. Para ello se produce una diferenciación de servicios y tráfico que permite que cuando la información es catalogada como crítica se establezca un canal dedicado que aumente las condiciones de entrega de las alarmas o mensajes.

Por otro lado mediante redundancia en los nodos y en las redes de comunicación se establece una línea de recuperación en caso de caída de la infraestructura de comunicación principal.

3.4.2.4 Disponibilidad

La disponibilidad del sistema tiene que plena en todo momento pues se afectan las comunicaciones del hospital. En caso de emergencia, la arquitectura debe contar con mecanismos de respaldo y autoconfiguración.

3.4.2.5 Mantenibilidad

El mantenimiento del sistema de comunicaciones consiste en comprobar que las redes están operativas y que los sistemas se pueden comunicar entre sí. Para ello se puede controlar de forma remota que componentes están en funcionamiento y cuáles no. Las tareas de mantenimiento se llevarán a cabo constantemente, es decir, mediante una gestión de equipos se podrá conocer el estado de los dispositivos, si está operativo o no.

3.4.2.6 Portabilidad de Datos

La portabilidad del sistema depende de los dispositivos o sensores que se encargan de las medidas de las constantes. Por otra parte la arquitectura debe soportar el empleo de distintos formatos y software médicos para facilitar la interoperabilidad con otros sistemas de comunicaciones de otros hospitales.

3.4.3 Otros requisitos y Restricciones

Podría incluirse como requisito adicional la compatibilidad de las redes y las emisiones de los componentes por radiofrecuencia con la legislación de emisiones en centros hospitalarios. De esta manera se crea una arquitectura de comunicaciones no perjudicial para la salud y no intrusiva para los pacientes.

Otro factor importante es que deben mantenerse las limitaciones al trato de información médica en entornos sanitario.

3.5 Adaptabilidad de Arquitecturas Hospitalarias

El avance de la computación ubicua y los servicios inteligentes de nueva generación orientados a entornos médicos plantean la necesidad de adaptar las arquitecturas hospitalarias existentes dentro de un contexto tecnológico común. En estos casos los requerimientos de un servicio no deberán ser impuestos por la red, sino por las propias características del servicio. De la misma manera estos servicios y el dinamismo con que sean desplegados y consumidos, deberán ser soportados en situaciones en donde las capacidades de las infraestructuras de red instaladas sean superadas.

El análisis de una arquitectura de red hospitalaria no se limita al soporte de los servicios que pudieran ser desplegados en la plataforma sino también en la adecuación de esta arquitectura al entorno en donde está desplegada y a las situaciones o eventos

específicos que la afecten. Este entorno puede ser tanto dinámico como estático, ya que diversos factores como emergencias, catástrofes, daños en equipos etc., pueden afectar la manera en que la información médica es distribuida desde los pacientes hacia el personal médico y viceversa.

Un entorno hospitalario en principio presenta características relativamente estáticas, en donde se conocen a priori cuáles son los servicios que han de estar disponibles y las consecuentes exigencias de cada uno. Pero como se mencionó anteriormente pueden existir eventos que se traducen en picos de tráfico que pudieran sobrecargar los servicios que son ofrecidos en la plataforma hospitalaria.

Está claro que no es viable el diseño de una arquitectura de servicios hospitalarios estática. Por lo tanto, para completar la flexibilidad, dinamismo y robustez de un sistema de este tipo, servicios estáticos deben ser sobrepuestos sobre arquitecturas de red dinámicas. Estas infraestructuras de red dinámicas deben aprovecharse de las infraestructuras estáticas ya existentes, para que se cumplan los requisitos exigidos por los procesos hospitalarios.

En este contexto de definición de arquitectura hospitalarias, se han analizados tecnologías como P4P[14] y EPC[15], desde el punto de vista de servicios, para que puedan sentar las bases de una colaboración entre redes y servicios hospitalarios dinámicos.

Finalmente algunos argumentos expuestos en este capítulo se han identificado como resultado de análisis realizados en proyectos como mIO!, CISVI y T2C2, que aunque no han sido enfocados específicamente en arquitecturas hospitalarias, si permiten obtener una concepto general relativos a los servicios y sus requisitos.

4 Propuesta de Arquitectura General

Las arquitecturas de red son en concreto la definición de las tecnologías utilizadas para proveer un servicio y la red en la cual se ofrecerá. Las arquitecturas de red abordan los problemas de una manera global, a la vez que ayudan a planificar, evaluar y seleccionar los distintos componentes necesarios para que un sistema funcione correctamente, además de las consecuentes necesidades de interoperabilidad que puedan surgir en la implementación.

Dependiendo del tipo de servicio, sus características y complejidad, se pueden utilizar una o varias arquitecturas de red que provean una solución específica para cada capa que lo componga, por lo tanto, las arquitecturas de redes son necesarias ya que regulan el diseño, construcción, administración y operación de una red de comunicación. Este documento se enfoca en contrastar el estado del arte de las tecnologías telemáticas orientadas a sistemas sanitarios, proponer y diseñar una arquitectura de comunicación hospitalaria que soporte múltiples servicios heterogéneos, y por último verificar su viabilidad mediante su desarrollo.

La arquitectura propuesta en este trabajo está enfocada en resolver los casos de uso mencionados en el punto 3.2, haciendo especial énfasis en los distintos elementos que son necesario, y funcionalidades que deben ser definidas. La arquitectura también es producto de las contribuciones hechas en los proyectos CARDEA[23] y CARDINEA[24]. Uno de los problemas principales es cómo mejorar una arquitectura hospitalaria, a la vez que se unifican los servicios médicos existentes y los datos provenientes de los sensores de constantes vitales. El sistema también deberá proveer mensajería instantánea con el fin de mantener al personal médico acerca de la evolución de los pacientes, y en caso de una emergencia tener la capacidad para priorizar los datos que deben ser enviados a través de los canales de comunicación. Otro de los aspectos primordiales es dar soporte a capas de software superiores que podrían agregar una inteligencia al sistema en general. Ya que existen diversos estudios [16][17] que se dedican a definir el contexto y a mejorar por las capacidades generales de los sistemas hospitalarios, la arquitectura propuesta se enfocará en identificar las necesidades de comunicación y provisión de datos que pudieran necesitar estas capas superiores. Con esto se intenta abstraer la complejidad de la red y cumplir con los requisitos mencionados en el punto 3.4.

Uno de los factores que afecta el diseño general de la arquitectura de red es el número de plantas y camas en un hospital y su distribución, ya que de ello depende la topología de red. Para que un sistema hospitalario sea escalable y dinámico no basta con agregar componentes a la vez que aumentan los requerimientos, sino que es necesario desde antes realizar una planificación sobre cómo irán colocados los nodos

de manera que los servicios que han de ser desplegados no sean afectados por la sobrecarga de los sistemas de comunicación. En la figura 8 se muestra una arquitectura física general que engloba los elementos involucrados en la transmisión de datos.

Describiendo de manera más específica ya que en cada cuarto de un hospital existe, como mínimo, un paciente, es necesario contar con un dispositivo que se encargue de concentrar todos los sensores de constantes vitales. A este dispositivo se le llamará iBed. Además deben existir elementos que se encarguen de monitorizar la localización de todo personal que esté en el cuarto, por lo que en el caso específico de esta arquitectura se utilizarán antenas y sensores RFID. Resumiendo un iBed será el elemento o nodo básico del sistema ya que concentrará los datos provenientes de sensores en cada cuarto. El iBed estará encargado además de transmitir toda la información hacia distintos nodos de la red, utilizando SIP para la señalización y el protocolo más conveniente (según el caso) para la transmisión de los datos. Esta separación entre señalización y datos ofrece una flexibilidad al sistema de manera que se cumple uno de los principios básicos de diseño en la ingeniería telemática.

Los iBed se conectan de manera física con un tipo especial de servidores llamado servidores intermedios, los cuales su funcionamiento será explicado luego detalladamente. Esta conexión se realiza mediante enlaces Ethernet, pero además plantea la utilización de una conexión Wi-Fi para asegurar la fiabilidad del sistema.

El propósito principal del servidor intermedio es de soportar múltiples nodos (ya sean iBed o dispositivos móviles) por cada planta. Para ello se realiza un pre-procesamiento en la capa de aplicación, o un encaminamiento a nivel de transporte para cada uno de los mensajes recibidos. Adicionalmente los servidores intermedios permiten monitorizar el estado de cada uno de los nodos, y contribuyen a prevenir una sobrecarga del servidor central (abstrayéndose de la topología), con lo que se agrega una característica de descentralización al sistema. El servidor central, que es definido así por su funcionalidad, más no por ser un único ente, será el destinatario final de todos los datos generados en el ambiente médico. El agregar una separación entre servidores intermedio y servidor central agrega una jerarquía a la red, que facilita las labores de gestión de todo el sistema en general. Otra de las ventajas que aporta este enfoque es la capacidad de procesamiento alarmas que pueden ser generadas en cada cuarto sin necesidad de interactuar con el servidor central.

El servidor central, cuyas funciones serán explicadas con detalle posteriormente, es el elemento principal de la arquitectura y es donde se guardan y procesan todos los datos que son capturados por los sensores. Además se envían/reciben alarmas y se agrega el soporte primario a la existencia de servicios heterogéneos. Estos servicios

heterogéneos serán el resultado de la combinación de distintos eventos, alarmas y preferencias definidas según las necesidades específicas en el entorno.

La comunicación física entre distintas plantas y el servidor central se establece inicialmente mediante enlaces de fibra óptica para asegurar el aislamiento eléctrico, con lo cual se evitan futuros problemas de pérdida de paquetes y daños en interfaces físicas. Este tipo de enlace también permite soportar ráfagas de datos que pudieran deberse a situaciones de emergencia, en donde la cantidad de datos médicos se incrementa rápidamente. Así mismo, la vida de un paciente podría ponerse en peligro en el caso en que los datos de constantes vitales o de mensajes de control de dispositivos médicos sean descartados o retrasados. Este tipo de enfoque aunque se ha abordado inicialmente de una manera un poco “práctica” tiene otra justificación ya que la capacidad existente en estos enlaces troncales incidirá directamente en los tipos de servicios que serán soportados por la arquitectura. Al mismo tiempo la corrupción de datos o transmisión errónea de paquetes es una característica inaceptable debido a los resultados que los servicios pudieran presentar.

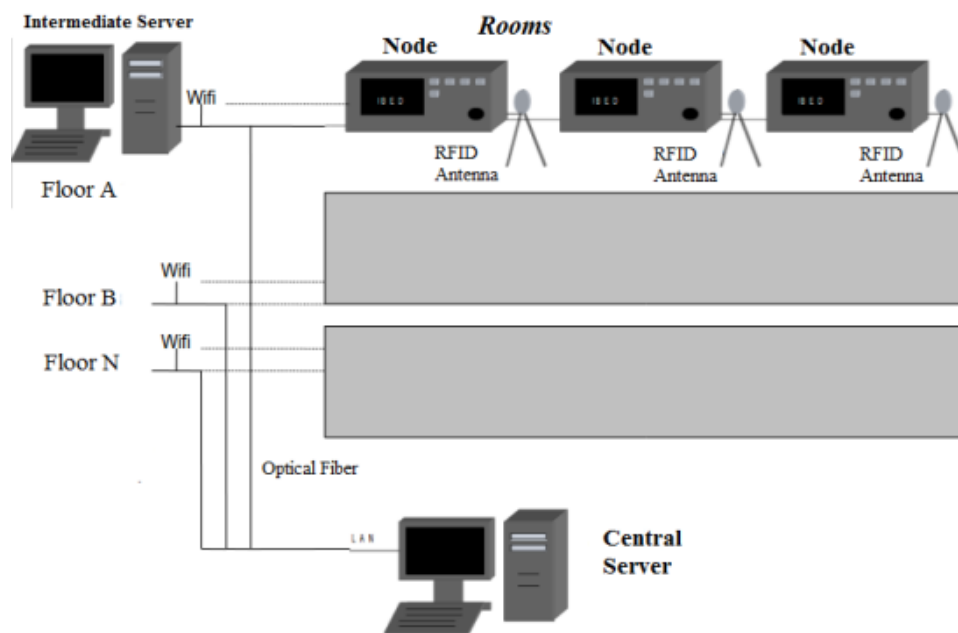


Figura 8. Arquitectura General Propuesta

4.1.1 Arquitectura de Nodos

La arquitectura de nodos está dividida en distintas capas y cada una de ellas implementa una funcionalidad específica. La primera capa es la capa de acceso a la red, la cual dependiendo del caso será Ethernet o Wi-Fi. La Interfaz de Comunicación está formada por dos elementos: un agente de usuario SIP y un módulo basado en el framework Mirth. El agente de usuario SIP (UA) tiene la función de proveer la gestión

de sesiones dentro del nodo, además de proveer movilidad mediante el registro de una dirección SIP-URI previamente asignada. El UA se encargará de comunicarse con un servidor Proxy-SIP localizado en el servidor intermedio. Otra de las características del UA, es que ese encargará de negociar las características de la sesión de datos que será transmitida hacia el servidor intermedio.

El módulo Mirth es el encargado de procesar los datos provenientes de los sensores y transmitirlos hacia el destino. Este módulo se encarga de analizar, y codificar si es el caso los mensajes HL7 e informar a otros elementos existentes en la arquitectura sobre las características de los datos adquiridos.

Existe otro módulo de Control que se encarga de realizar toda la lógica necesaria en caso de que ciertos parámetros sean recibidos, o ciertos requerimientos de comunicación se necesiten. De esta manera cuando un dato está disponible para enviarse, el módulo de Control ordena, al agente de usuario SIP, que cree una sesión de datos basada en atributos predefinidos que serán explicados posteriormente. Estos atributos podrán ser estáticos o dinámicos, dependiendo de en qué contexto se esté realizando la transmisión de datos. Como siguiente paso el módulo basado en Mirth, construye o genera un canal de datos con las propiedades y protocolos que sean óptimos en ese momento. Por este canal de datos se enviarán todos los mensajes codificados en HL7, incluyendo la información proveniente de distintos equipos médicos. Adicionalmente la introducción de tecnologías que garanticen servicios y latencia como Diffserv dentro de los paquetes IP supondría un nivel de garantía de calidad de servicio superior. El Módulo de Control tendrá la función de controlar a nivel de IP la utilización de estos otros tipos de parámetros

Otro módulo que estará disponible en los nodos, y por lo tanto en la arquitectura general OGSi es el módulo RFID. Éste se encargará de detectar las etiquetas RFID, la posición de los pacientes/personal médicos, y la monitorización de fármacos. El modo de comunicación con los servidores de la arquitectura general se realiza mediante los mismos criterios de transmisión que la comunicación de sensores, por lo tanto se separará las sesiones de los datos. Una de las características principales de este diseño es que aporta una flexibilidad y una unificación, ya que distintos elementos funcionales se apoyan en un único agente de usuarios SIP, con lo que se abstrae la implementación de la pila de protocolos a un nivel mucho más simple.

En la figura 9 se muestra la arquitectura de un nodo hospitalario. En él, cada una de las funcionalidades está basada en el framework OSGi, con lo cual se asegura la interoperabilidad y la reutilización de componentes. Está claro que adicionar esta framework puede traducirse a primera vista en una complejidad en la implementación, pero las ventajas que son aportadas y que ya han sido descritas en los capítulos

anteriores, apoyan su utilización. Este diseño, aunque no aparece en la figura, puede soportar la adición de otros tipos de tecnologías ya que la inclusión de las interfaces de comunicación con protocolos como TCP/IP lo permite. De esta manera elementos que no puedan ser incluidos por razones tecnológicas, de arquitectura, o de situación física/lógica dentro del framework OGSi podrán ser reutilizados.

La arquitectura es capaz también de abstraer otros elementos que sean fuentes de datos y que no estén físicamente conectados al nodo. Esto mediante la inclusión de distintos APIs, con lo cual se asegura la interoperabilidad en todas las capas de comunicación. Con este modelo se permite finalmente la abstracción de la red, y una homogenización de los servicios/funcionalidades que puedan ser provistas por otros elementos de otras redes que no estén basadas en Ethernet o Wi-fi.

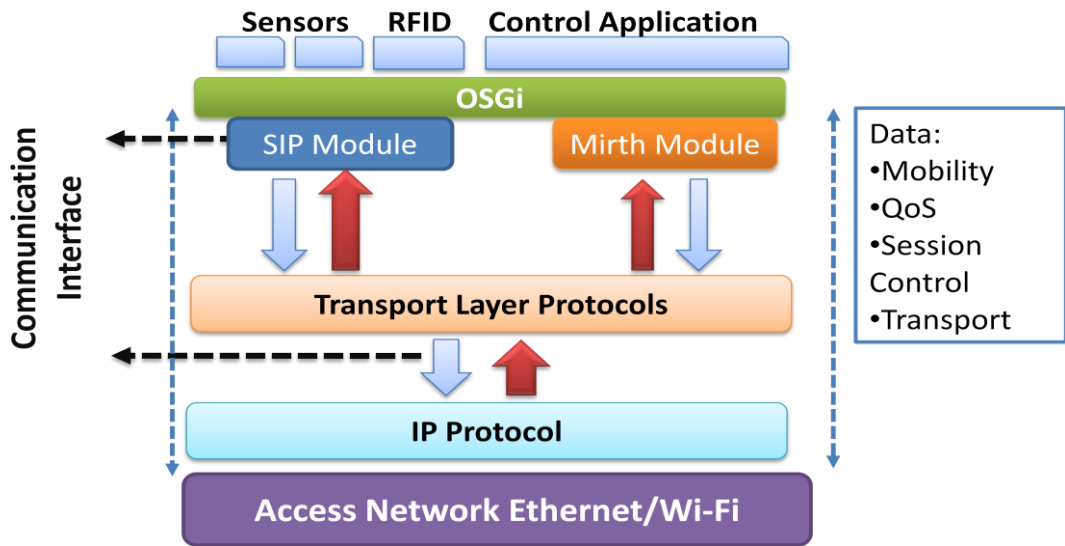


Figura 9. Arquitectura de Nodos

A continuación se resume en una tabla las funcionalidades que podrán ser soportadas por cada uno de los nodos. Dependiendo del caso y de las capacidades tecnológicas del mismo podrán implementarse o no en su totalidad.

Funcionalidades	Soporte mediante módulos o “bundles”
<p>Comunicación con servidores para gestión de información</p> <p>a) Punto a Punto y Multipunto</p> <p>c) Movilidad</p> <p>d) Presencia</p> <p>e) Calidad de Servicio</p> <p>f) Resistencia a fallos</p>	<p>Mediante SIP, se establecen las sesiones necesarias para la comunicación a través de la red IP.</p> <p>a) Sesiones SIP</p> <p>b) Envío de alarmas generales a dirección multicast o por varias sesiones SIP</p> <p>c) Autoconfiguración mediante usuarios SIP</p> <p>e) Calidad dependiente del servicio</p> <p>f) Redundancia de redes (Ethernet y wifi) y sistema petición-respuesta de SIP.</p>
<p>Envío de constantes y medidas en formato HL7</p> <p>a) Recogida de datos en crudo</p> <p>b) Codificación/decodificación en HL7</p> <p>c) Transmisión a y desde los servidores</p>	<p>Combinación de SIP y Mirth, para añadir las funcionalidades de SIP y aprovechar la creación de mensajes HL7 por parte de Mirth.</p> <p>a) Interfaz genérico de sensores</p> <p>b) Librería Mirth de codificación y decodificación</p> <p>c) Sesiones SIP + Transmisión de documentos HL7</p>
<p>Detección de paciente.</p>	<p>RFID, antena colocada de forma que controle la cama del paciente.</p> <p>El control de presencia de SIP permite la distinción de pacientes.</p>
<p>Visualización de alarmas en pantalla.</p>	<p>Capacidad de interacción con dispositivos como pantallas LCD etc.</p> <p>Envío de Alarmas mediante SIP desde los servidores a los nodos</p> <p>Uso de Presencia para identificar los nodos</p>
<p>Diferenciación de la importancia de cada dato enviado.</p> <p>a) Decisión de los recursos que se van a emplear en su envío.</p>	<p>Programando los DiffServ de IP para proporcionar diferentes valores a cada mensaje. Y utilizando también SIP para el manejo de sesiones según la cantidad de tráfico.</p>
<p>Detección de los medicamentos adecuados a cada paciente.. Controlar que el medicamento está prescrito para el paciente.</p>	<p>Detección de los medicamentos adecuados a cada paciente. Mediante etiquetas RFID.</p> <ul style="list-style-type: none"> - Detección de paciente. - Detección del medicamento - Correspondencia entre los dos anteriores
<p>Adición de nuevos servicios y sensores</p>	<p>Bundles OSGi que se integran sobre Java, creando nuevos servicios y relaciones</p>
<p>Control de estado de los sensores conectados a los nodos.</p>	<p>Mediante la comunicación con los sensores en las diferentes medidas se captaría si algún sensor está en mal estado y se comunicaría al servidor intermedio para enviar una alarma.</p>
<p>Detección de visitas y consultas</p>	<p>Lector RFID con etiquetas personales que deberían llevar el personal.</p>
<p>Movilidad del nodo y control de presencia</p>	<p>Mediante las facilidades que aporta SIP. Acceso a la red en puntos distintos. Control de presencia de usuarios</p>

Tabla 1 – Funcionalidades de Nodos

4.1.2 Arquitectura de Servidor Intermedio

La arquitectura de los servidores intermedios se ha diseñado tomando en consideración la misma pila de protocolos que los nodos. La arquitectura sin embargo cuenta con una flexibilidad mayor debido a las propias características del dispositivo donde será ejecutada. Mientras que los nodos, son pensados como equipos móviles y/o con limitada capacidad de procesamiento, los servidores intermedios tendrán por definición una capacidad superior. Además de esto deberán implementar distintas funcionalidades de control y monitorización de los equipos que correspondan a la propia planta para la cual han sido asignados.

A nivel de comunicación un servidor intermedio, tendrá un agente SIP, el cual será un SIP Back-2-Back User Agent (B2BUA) que tendrá más capacidad para procesar tanto mensajería SIP, como tráfico relacionado con datos prioritarios. Existirá dentro de él también un módulo Proxy, que mantendrá la lista de todo los SIP UA registrados, además de las sesiones establecidas y los recursos de red disponibles en cada momento.

La plataforma de funcionalidades está basada, como los nodos, en OSGi, y se le podrán expandir los módulos incluidos inicialmente. El servidor intermedio también es capaz de procesar toda la mensajería HL7 ya que debe poder discernir entre los distintos tipos de datos que han sido generados por los sensores. Los servidores tienen una módulo llamado OCP (Open Context Platform) en la capa de aplicación, el cual tiene la función de procesar la información y reaccionar ante posibles eventos.

A continuación se muestra una tabla con las funcionalidades:

Funcionalidades	Soporte mediante módulos o “bundles”
<p>Comunicación con servidores y nodos para gestión de información</p> <ul style="list-style-type: none"> a) Punto a Punto b) Multipunto c) Movilidad d) Presencia e) Calidad de Servicio f) Resistencia a fallos 	<p>Mediante SIP, se establecen las sesiones necesarias para la comunicación a través de la red IP.</p> <ul style="list-style-type: none"> a) Sesiones SIP b) Envío de alarmas generales a dirección multicast o por varias sesiones SIP c) Autoconfiguración mediante usuarios SIP e) Calidad dependiente del servicio f) Redundancia de redes (Ethernet y Wi-fi), diferentes fibras ópticas por cada planta del hospital. Y sistema petición-respuesta de SIP.
<p>Envío de constantes y medidas en formato HL7</p> <ul style="list-style-type: none"> a) Codificación/decodificación en HL7 b) Transmisión a y desde los servidores 	<p>Combinación de SIP y Mirth, para añadir las funcionalidades de SIP y aprovechar la creación de mensajes HL7 por parte de Mirth.</p> <ul style="list-style-type: none"> a) Librería Mirth de codificación y decodificación b) Sesiones SIP + Transmisión de documentos HL7 => <ul style="list-style-type: none"> a) Stream de datos <-> Stream de documentos HL7 b) Dato asíncrono == documento HL7 c) Grupo de datos == documento HL7
<p>Almacenamiento de información médica y situación de pacientes.</p> <ul style="list-style-type: none"> a) Estructuración de historiales médicos. b) Almacén de datos capturados por los sensores. 	<p>El bundle OCP de OSGi se encarga de gestionar el contexto y toda la información obtenida.</p>
<p>Visualización de alarmas en pantalla y detección de situaciones peligrosas.</p>	<p>Mediante la información del bundle OCP y las relaciones OSGi entre bundles se detectan posibles crisis y acciones a ejecutar</p>
<p>Diferenciación de la importancia de cada dato enviado.</p> <ul style="list-style-type: none"> a) Decisión de los recursos que se van a emplear en su envío. 	<p>Utilización de DiffServ e IP para proporcionar diferentes valores a cada mensaje. Y utilizando también SIP para el manejo de sesiones según la cantidad de tráfico.</p>
<p>Control de stock de los medicamentos.</p> <ul style="list-style-type: none"> a) Identificación del medicamento. b) Registro de entrada/salida del medicamento en el sistema. c) Visualización de alarma de falta de stock si fuera necesaria 	<p>Mediante etiquetas RFID que leerán los receptores RFID y a través del bundle RFID se identifican. Y mediante el bundle OCP de OSGi se detectan las deficiencias de stock.</p>
<p>Adición de nuevos servicios y sensores</p>	<p>Bundles OSGi que se integran sobre Java y que se comunican entre sí para establecer nuevos servicios.</p>
<p>Distribución de datos (historial ...)</p>	<p>La utilización de la red IP, ya sea mediante wifi o ethernet.</p>
<p>Control de personal</p> <ul style="list-style-type: none"> a) Exposición del personal a agentes nocivos b) Posible control de posicionamiento y trabajo. 	<p>Mediante etiquetas RFID asignadas al personal, la información la almacena y gestiona el bundle OCP.</p>
<p>Gestión de los equipos de su planta predeterminada.</p> <ul style="list-style-type: none"> a) Control de actividad y estado de sensores y equipos de su planta. b) Activación de alarma por inactividad si fuera necesario. 	<p>Mediante la comunicación con los nodos de las cabeceras obtiene información sobre los sensores activos y lo almacena en el contexto para una posible activación de alarma distribuida por ESB.</p>

Tabla 2 – Funcionalidades de Servidor Intermedio

4.1.3 Arquitectura de Servidor Central

El servidor central es el elemento principal de la arquitectura. Ya que se encargará de reaccionar ante los eventos y las posibles situaciones de riesgo que ocurran. Además de las funciones de servidor intermedio, el servidor Central tiene las siguientes funciones.

Funcionalidades	Soporte mediante módulos o "bundles"
Detección de servidores inactivos.	Mediante bundle SIP y el control de presencia a través del cable de fibra óptica por el cual está conectado con los demás servidores.
Cambiar su rutina de trabajo si se cae un servidor intermedio para realizar el trabajo del servidor caído.	Comunicación por red, con utilización de programación en Java para cambiar la rutina de trabajo en casa de caída.
Gestionar todos los equipos del sistema. a) Control de actividad y estado de todos los sensores y equipos del sistema. b) Activación de alarma por inactividad si fuese necesario.	Mediante la comunicación con los servidores intermedios obtiene información de todos los equipos y almacena su estado.

Tabla 3 – Funcionalidades de Servidor Central

4.1.4 Arquitectura y Topología de Red

La arquitectura de red define la manera en que los distintos elementos de red ya sean nodos, servidores intermedios o servidores centrales son desplegados en un ambiente hospitalario. Este despliegue aunque físicamente pueda tener múltiples opciones, lógicamente tendrá el esquema que se ha definido en la figura 10. Los tres componentes principales antes listados basan todas sus interfaces de comunicación en los mismos protocolos, pero existe una pequeña diferencia entre los servidores intermedios y el central. Como resultado de las características del propio canal de comunicación (fibra óptica), las exigencias de ancho de banda y la capacidad de procesamiento, el tráfico entre los servidores puede ser gestionado mediante protocolos más fiables. Además de esto, debido a la inclusión previa de tecnologías de reserva de tráfico a nivel IP, es necesario dotar a los servidores de mecanismos para que sean conscientes del tipo de tráfico que se está enviando/recibiendo a nivel IP.

Los datos generados en ambientes hospitalarios causan que los la arquitectura pueda soportar las exigencias de fiabilidad e integridad, además de otras mencionadas posteriormente. Adicionalmente en situaciones de emergencia se añaden requerimientos aún mayores a la propia red, como resultado de esto los servidores

centrales tendrán la función de verificar mediante mensajes de estado, la disponibilidad, y carga de trabajo de los intermedios. En el caso de que un servidor secundario falle, los nodos contarán con un mecanismo que detectará el evento y redireccionarán el tráfico hacia el servidor central, además de renegociar las sesiones catalogadas como críticas mediante el protocolo SIP. Este mecanismo de renegociación es controlado por funcionalidades implementadas mediante bundles OSGi que estarán en capas superiores.

Además de ello un servicio y los datos asociados a él podrán auto-configurarse en el caso en que ocurra un fallo de red o cuando elementos lógicos existentes en la **arquitectura central del sistema hospitalario** (servidores intermedios y servidor central) lo decidieran. Ejemplo de estos eventos podrían que desencadenen decisiones podrían ser: alto tráfico, fallos puntuales de equipos, emergencias etc. Esta modularidad existente en el diseño permite una abstracción de los servicios básicos existentes en la red y un estado de conocimiento de la misma, con lo que se asegura la fiabilidad del sistema.

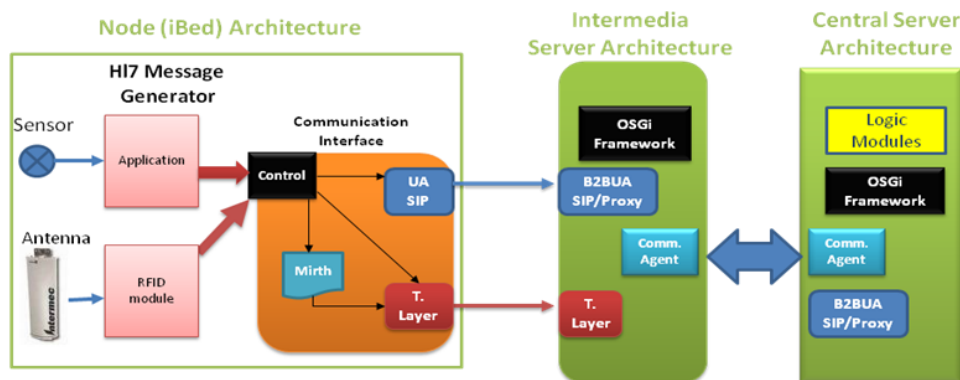


Figura 10. Arquitectura Global de Red

4.1.5 Modelo de Datos

Como se mencionó con anterioridad factores como ancho de banda y latencia, tienen un alto impacto en ambientes médicos[20]. Por lo tanto es necesaria dentro de una arquitectura de servicios hospitalarios una clasificación de datos según su tipo. Además de ello esta clasificación deberá ser condicionada al dinamismo de la red y a la situación en la cual se esté dando la transmisión de información.

El modelo de datos tiene una relación directa con la información sobre constantes vitales relacionada con los pacientes. Esta información deberá ser registrada en la plataforma de servidores y estar disponible para su posterior procesamiento. Definir un modelo de datos permite una forma fiable, segura, y óptima de transmitir los datos a través de la arquitectura de comunicación que se está desarrollando.

Cada tipo de datos necesita unos parámetros de calidad de servicios específicos. Al mismo tiempo debe definirse tráfico tolerantes (tolerant) o no tolerantes (non-tolerant). El primero de ellos puede ser adaptado a la calidad del enlace existente en ese momento mientras que el segundo no, por lo que se le debe dar un trato más expedito.

En los datos médicos la arquitectura diferencia entre datos en tiempo real y datos de no tiempo real. El primero es definido como no tolerante y tiene que cumplir ciertos requerimientos como lo son: bajo jitter, baja latencia, alto procesamiento y eficiente gestión de acuses de recibo (ACK). Dependiendo de las características del mensaje, los recursos de red son reservados o no. En esta arquitectura se definen tres tipos de tráficos: tráfico urgente (urgent traffic), tráfico importante (important traffic) y tráfico de mejor esfuerzo (best effort). Otra clasificación puede ser hecha mediante la diferenciación de datos que son pueden ser enviados por paquetes datos (tipo mensajes) por ráfagas (tipo audio, video). El tráfico por ráfagas tiene requerimientos especiales con el fin de mantener la tasa de transmisión/recepción por unidad de tiempo pero no principalmente la fiabilidad.

En la siguiente tabla se clasifican los distintos tipos de datos médicos que son transmitidos sobre una red hospitalaria, y la calidad de servicio relacionada. Esta tabla facilita una correlación con la información y los requerimientos de los datos provenientes de sensores u otros dispositivos médicos.

Traffic Type	QoS			
	Delay	Jitter	Bandwidth	Reliability
Urgent traffic (non-tolerant)	Very low	Low	High	High
Important traffic (non-tolerant)	Low	Low	High	Medium
Important traffic (tolerant)	Medium	High	Medium	High
Best Effort (tolerant)	High	High	Low	High

Tabla 4 – Tipo de Tráfico vs Calidad de Servicio

- Urgent Traffic (non-tolerant): datos automáticos generados por el estado crítico de un paciente y que requieren una atención inmediata. Algunas alarmas y fallas en equipos médicos pueden ser clasificadas en esta categoría además de imágenes o video.
- Important Traffic (non-tolerant): datos capturados manualmente a petición del personal médico. Por ejemplo: frecuencia respiratoria, drenaje, clasificación diurética, volumen pulmonar etc.
- Important Traffic (tolerant): data proveniente de espirómetros, nivel de glucosa, u otros datos provenientes de sensores de temperatura, presión de sangre, pulso etc. Ciertos datos provenientes de equipos médicos pueden ser catalogados aquí, en especial en situaciones de emergencia o riesgo.
- Best Effort (tolerant): este grupo incluye todos los datos, como archivos, historial médico o cualquier tráfico que no tiene altos requerimientos y que no afecta directa o indirectamente el estado de un paciente o personal médico.

Una de las ventajas del modelo de datos definidos, es que permite la gestión de sesiones y de recursos de red según atributos predefinidos. En el caso particular en el que un nodo quiera enviar un dato hacia cualquier otro elemento del sistema, el Módulo de Control existente se encargará de todo el proceso. Este módulo informará al agente SIP, acerca de los requerimientos de comunicación. Como resultado el módulo SIP negociará una sesión con el Servidor Intermedio utilizando protocolos como SDP, a la misma vez que se toma en consideración los recursos existentes en la red, además de las características (soporte de Diffserv etc.) de los dispositivos de comunicación intermedios (switches, routers etc). Una vez que la sesión ha sido establecida el módulo Mirth transmitirá los datos capturados por los sensores a través de un canal especial creado con los parámetros óptimos.

Entre las ventajas listadas por el mecanismo descrito anteriormente están que múltiples protocolos se podrán utilizar para múltiples tipos de datos. Por ejemplo si un canal de datos es generado, el modulo Mirth podrá escoger entre RTP o UDP, con el fin de prevenir sobrecarga en la red y los sistemas en general. Sin embargo, si el tipo de datos no demanda una alta QoS, pero sí fiabilidad, TCP puede ser utilizado. Sistemas similares[21][22] ya han sido analizados, pero el propuesto en este trabajo aborda el tema desde un punto de vista más específico.

Finalmente en el Modelo de Datos, el módulo o “bundle” de Control tiene dos tareas cruciales: la gestión de correspondencia entre tipos de datos y calidad de servicio asignado, y el soporte para interoperabilidad entre las distintas capas de comunicación, como lo serían SIP-UA, y Mirth.

5 Desarrollo de Arquitectura

El desarrollo de una arquitectura no es más que su implementación mediante la utilización de elementos y tecnologías reales. El desarrollo está basado en los conceptos previamente definidos, y las funcionalidades específicas que han sido abordadas en una arquitectura a alto nivel.

El desarrollo de la arquitectura propuesta está basado en los conceptos expuestos en el capítulo anterior, los cuales son producto de un análisis y estudio de la problemática hospitalaria con respecto al soporte de nuevas tecnologías, situaciones de emergencia, optimización de la red, interoperabilidad entre equipos y despliegue de una nueva generación de servicios hospitalarios heterogéneos y colaborativos.

El desarrollo a continuación es parte de los trabajos previos realizados en los proyectos CARDEA y CARDINEA, en donde se definen una serie de requisitos y la necesidad de un nuevo enfoque con respecto a un sistema de comunicación hospitalario del futuro.

Una primera prueba de concepto fue desarrollada en el Hospital Gregorio Marañón, en donde se han adaptado las tecnologías RFID para detección de alarmas y pacientes, además de la implementación de alarmas mediante el uso de SIP.

En una segunda fase se ha trabajado en extender los servidores intermedios y servidores centrales, así como la mensajería, negociación de sesiones SIP, y optimización de envío de datos. Esta segunda fase está planeada para ser implementada en el Hospital de Tarrasa, Cataluña. De los datos resultantes de esta fase se obtendrán importantes requisitos concernientes a la interoperabilidad con equipos médicos reales, y que estén en producción, así como el comportamiento de la red ante eventos que pudieran afectar la fiabilidad y disponibilidad del sistema en general.

5.1 Consideraciones de Diseño

Para el desarrollo de un prototipo y la extensión de funcionalidades se adaptará el sistema en una planta de un hospital con cuatro habitaciones por lo que sólo se hará uso de un servidor intermedio, que tendrá las funciones de proxy SIP, además de esto, se implementará un servidor central el cual contará con todos los módulos de comunicación (Mirth y SIP) para la adaptación de datos.

Ya que la plataforma OSGi es el elemento que da soporte a toda la arquitectura definida, se utilizará la variante Knopflerfish[25] que es de Código Abierto. Esta variante permite también el acceso remoto a la plataforma, lo cual es importante para

establecer una gestión de la arquitectura. Al mismo tiempo Knoplerfish permite una mayor facilidad de interacción entre elementos o bundles existentes en el sistema.

5.2 Arquitectura Detallada de Módulos

Para la validación y desarrollo de la arquitectura propuesta se han implementado los módulos de comunicación SIP y Mirth que permitirán alimentar otros módulos que permitan gestionar y agregar el dinamismo a la arquitectura. De esta manera se verifica el soporte de múltiples servicios heterogéneos que pudieran ser soportados por la arquitectura propuesta en este documento. En los Proyectos CARDEA y CARDINEA se han desarrollado otros módulos como lo son, OCP (Open Context Platform) e Interfaces Web, pero su contenido y descripción escapan de los objetivos planteados por este trabajo. Finalmente cada módulo se encarga de una funcionalidad distinta y muy definida.

5.2.1 Módulo SIP

Uno de los elementos más importantes del sistema de comunicaciones es el módulo SIP puesto que es el encargado de la transmisión de los datos entre los distintos elementos de red. Adicionalmente la integración de servicios SIP permite establecer comunicación entre el framework OSGi y nodos externos utilizados para la recepción de mensajes de alarma. El módulo o bundle SIP interactúa con los elementos OSGi donde se registran los servicios que están presentes en la arquitectura. Este servicio integra principalmente tres elementos de la arquitectura SIP:

Agente de usuario: Constituye el elemento principal de las comunicaciones SIP. El protocolo SIP es un protocolo de nivel de aplicación que utiliza la estructura cliente-servidor para realizar conexiones extremo a extremo. Los elementos situados en los extremos son los agentes de usuario (UA).

Servidor proxy: Un proxy es un elemento intermedio del proceso de comunicación SIP que se encarga de recibir peticiones SIP y reenviarlas hacia su destino. Aunque no es un elemento fundamental para desarrollar un sistema de comunicaciones, se ha añadido a la arquitectura para proveer encaminamiento y redirección de sesiones.

Servidor de registro: El servidor de registro es un elemento encargado de recibir las peticiones de registro SIP y su procesamiento, actualizando la información del agente de usuario (dirección, disponibilidad u otras preferencias) situado en el servicio de localización (SIP Location Service). El servicio de localización se implementa como una

tabla “Hashtable” donde se almacena la correspondencia entre la dirección URI de los usuarios y su dirección física.

La integración de SIP en un sistema OSGi genera dos ventajas fundamentales. En primer lugar se convierte cualquier servicio OSGi en un SIP UA “virtual”. De esta forma, un dispositivo SIP externo puede establecer comunicaciones utilizando este protocolo sobre un entorno de desarrollo OSGi. Por otra parte, se extiende OSGi para poder manejar un dispositivo SIP desde las aplicaciones instaladas en su framework. De esta forma se dota a los servicios desplegados en la plataforma OSGi de las características de movilidad, presencia y flexibilidad del protocolo SIP.

La arquitectura permite que los terminales móviles o PDAs, con capacidad de comunicaciones por SIP, se registren en el servidor de registro implementado en la plataforma. En este momento, las aplicaciones desplegadas en la arquitectura pueden hacer uso de la funcionalidad proporcionada por el bundle SIP, permitiendo conectarse y enviar mensajes al dispositivo usando la URI registrada.

Hay que tomar en cuenta que la funcionalidad SIP, estará implementada también en forma de elementos independientes de OSGi, ya que existen dispositivos móviles que por sus propias características y funciones, no compensan la inclusión de esta plataforma. En estos casos se incluirá un API, que ejecute las mismas funciones.

5.2.2 Módulo Mirth

Los nodos que toman las medidas funcionan con un software médico que gestiona la mensajería HL7. En el caso específico de esta arquitectura se ha elegido Mirth para realizar esta función. Como se mencionó anteriormente Mirth proporciona una interfaz para la creación de los mensajes que en el caso, además que soporta la adición de otros protocolos como HTTP, RSTP o FTP. Por lo que finalmente en la arquitectura se pueden aprovechar las ventajas que darían la utilización dual de ambos métodos de comunicación.

La combinación de SIP con Mirth es necesaria puesto que se soporta la utilización de formatos médicos para hacer más fácil la obtención de información y así generar los historiales electrónicos. Mirth es un ESB (Enterprise Service Bus) para entornos hospitalarios, en nuestro caso también se integra un bundle ESB para permitir el intercambio de información con aplicaciones ajenas a nuestra arquitectura.

5.2.3 Módulo de Control

Como se mencionó en el capítulo anterior, el módulo de Control, que es en principio un bundle OSGi, se encarga del análisis de los requerimientos de comunicación según los datos que se desean transmitir. Además tiene la función de renegociar sesiones y detectar si hay nuevos requisitos. Este módulo al igual que los demás, puede estar como un elemento independiente del sistema OSGi, lo cual lo hace independiente y adaptable.

5.3 Implementación de los Nodos

El sistema existente cuenta con 3 tipos de nodos, cada uno con una implementación de funcionalidad distinta. La diversidad en la implementación de nodos aporta una flexibilidad al sistema en donde se la arquitectura soporta tanto sistemas cerrados como abiertos.

5.3.1 Nodos Multifunción

Un nodo multifunción puede ser capaz de ejecutar varias tareas. Este tipo de nodos está en una habitación y ejecuta una plataforma OSGi Knoplerfish. Entre las características de software están:

- Sistema Operativo Ubuntu Linux 9.10 con JDK 6
- Knoplerfish v2.33
- Bundles SIP, RFID, Mirth y Bluetooth.

El nodo multifunción a ser implementado totalmente en código abierto es muy flexible y tiene interfaces con sensores inalámbricos que usen Bluetooth. Este nodo se caracteriza por ser muy versátil, y compatible con los nuevos estándares que puedan ser desarrollados. Otra de las ventajas está es que es fácilmente actualizable.

Entre las desventajas se encuentra que su utilización está limitada a tareas que no sean demandantes, y que no estén directamente relacionadas con la salud de un paciente, ya que por ser un ordenador común, no está certificado totalmente para ambientes médicos.

5.3.2 Nodos de Arquitectura Cerrada

Dentro de esta categoría se han incluido nodos cuya arquitectura no está disponible para modificación o análisis, ya sea porque el código no está disponible, por las limitantes de hardware, prohibición del fabricante, patentes etc. Estos tipos de dispositivos se presentan como alternativas cerradas y certificadas por fabricantes específicos.

Específicamente se han utilizado nodos llamados iBed. Los nodos iBed son soluciones de Telemonitorización Intrahospitalaria de arquitectura cerradas que permiten la integración con dispositivos biométricos de diagnóstico.

iBEDs TV



Formato	Pantalla táctil 15" o 17" con capacidad de comunicaciones IP (WiFi, Ethernet)
Descripción Funcional	Incorpora la misma funcionalidad del iBEDs nano, construida sobre sistema Operativo Windows XP Professional Su instalación puede realizarse a través de un flexo de brazo largo o integrada en la propia mesa del paciente
Comunicaciones	Integra comunicaciones inalámbricas, WiFi y Ethernet Comunicación serie del dispositivo integrada en el Flexo o carro Integra telefonía IP
Funcionalidades de valor añadido	Ejecución de cualquier aplicación o Historia Clínica informalizada Incorpora TDT y permite la distribución de contenidos IPTV, DVB, Telefonía IP

Solución iBEDs que complementa el uso profesional y el de ocio para el paciente, integrando comunicaciones, TDT, IP TV, DVB y reproducción de contenido diseñado ad hoc (formación, juegos, ...)

Figura 11. Características de un iBed

A los iBeds soportan distintos dispositivos gráficos y sensores biométricos como lo son:

- Monitores Multifunción, que permiten medir temperatura, SPO2 y Presión Sanguínea no Invasiva.
- Glucómetro
- Espirómetros

El sistema de comunicación de los iBed es cerrado y no existe posibilidad de modificación alguna de su funcionamiento, por lo que las funcionalidades antes listadas no pueden incluirse a nivel de nodo. Los iBed transmiten toda la información capturada por los sensores hacia un servidor "concentrador" utilizando un formato de datos propietario.

Para la inclusión de este tipo de sistemas cerrados dentro de la arquitectura propuesta se han definido una serie de mecanismos. El servidor concentrador enviará todos los datos provenientes de los iBED, hacia los servidores intermedios siguiendo el estándar HL7 en donde se analizarán los campos (sensor, paciente etc) más importantes incluidos en los mensajes, para luego construir una canal de comunicación

con las directrices explicadas anteriormente. Una vez hecho lo anterior, toda la información proveniente de sensores estará disponible dentro de la arquitectura general. El punto negativo es que no se podrán aprovechar todo el potencial que provee la arquitectura ni los futuros servicios que pudieran estar disponibles. Sin embargo la importancia de incluir sistemas cerrados dentro de un prototipo es que se comprueba la compatibilidad de la arquitectura con este tipo de soluciones, que están muy extendidas dentro de ambientes hospitalarios.

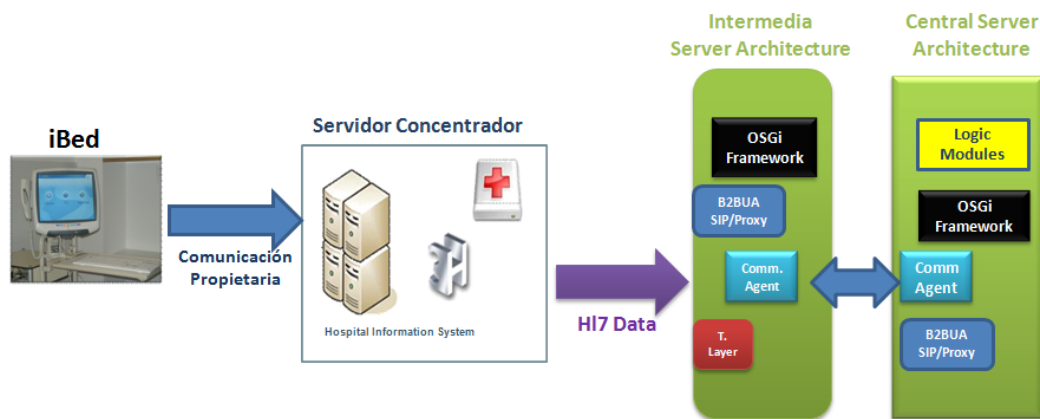


Figura 12. Interacción de Arquitectura con Sistemas Cerrados

5.3.3 Implementación en Nodos Móviles

Los nodos móviles se han implementado de dos tipos, pero ambos utilizando SIP como protocolo de comunicación con los otros elementos de la red.

Se ha diseñado un prototipo de cliente SIP para recepción de mensajes, en donde el personal médico recibirá alertas generadas por eventos que ocurran en el sistema, ya sea el estado crítico de un paciente, una baja en el stock de medicinas etc. El prototipo se ha programado en J2ME. La interfaz gráfica es simple, pero lo suficiente para proveer información útil.

Otra de las características es que el prototipo hace uso de un API genérico con lo cual la interfaz gráfica no se limitará a una sola implementación.

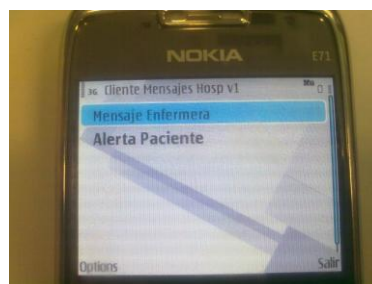


Figura 13. Lista de Mensajes Recibidos

El prototipo ha sido probado utilizando el Sistema Operativo Symbian, específicamente en un Nokia E71, que es catalogado como un teléfono de gama alta. Pero gracias a la portabilidad de Java podría ser implementado en otros teléfonos móviles.



Figura 14. Integración Cliente SIP en Teléfono Móvil

Se ha realizado otra implementación de un nodo móvil, esta vez en lectores RFID, que capturan la información de las etiquetas RFID que son registradas en por el sistema. A diferencia de la implementación anterior, estos tipos de dispositivos utilizan una versión de Windows CE que soporta también J2ME. Con este prototipo se verifica el soporte a distintos módulos que no estén relacionados explícitamente con arquitectura de red. En este caso, se ha hecho un desarrollo de captura de eventos basados en una plataforma de agentes llamada JADE, pero que a su vez utiliza las interfaces SIP provistas.



Figura 15. Integración de API SIP en Lector RFID

5.4 Implementación de Servidores

Los prototipos que validan la arquitectura en los servidores intermedios y el servidor central, están basados en tecnologías de código abierto. A continuación se detallan las características de ambos.

- Sistema Operativo Ubuntu Linux 9.10 con JDK 6
- Knoplerfish v2.33
- Bundles SIP, Mirth, OCP y RFID.

La arquitectura del prototipo y los elementos que se han implementado están disponibles en la siguiente figura:

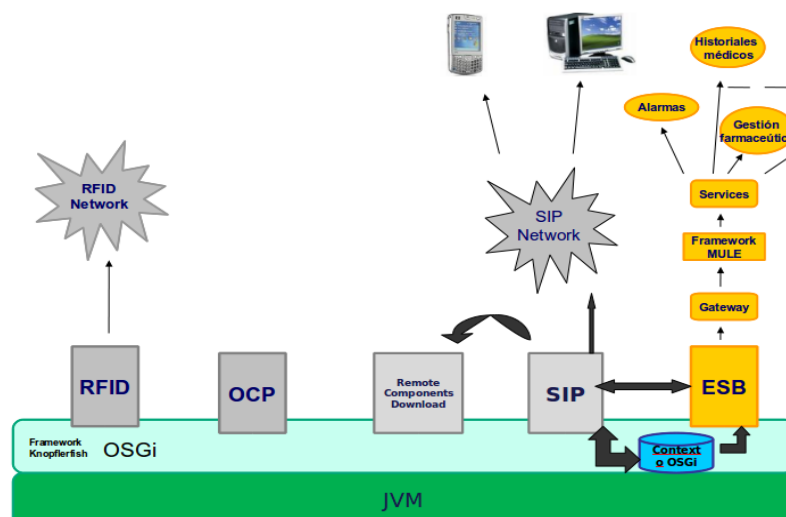


Figura 16. Arquitectura de Prototipo de Servidores

Existen diversos bundles [26] que han sido desarrollados por terceros, ya que la implementación de la arquitectura es el resultado del análisis y colaboración en varios

proyectos. Pero a su vez todos ellos se basan en la abstracción de la arquitectura de comunicación provista por las funcionalidades descritas en este trabajo.

La figura 17 muestra la interfaz de ejecución OSGi existente en un servidor intermedio, dedicado al procesamiento de paquetes SIP (además del servidor de registro) y la codificación/decodificación de mensajes HL7.

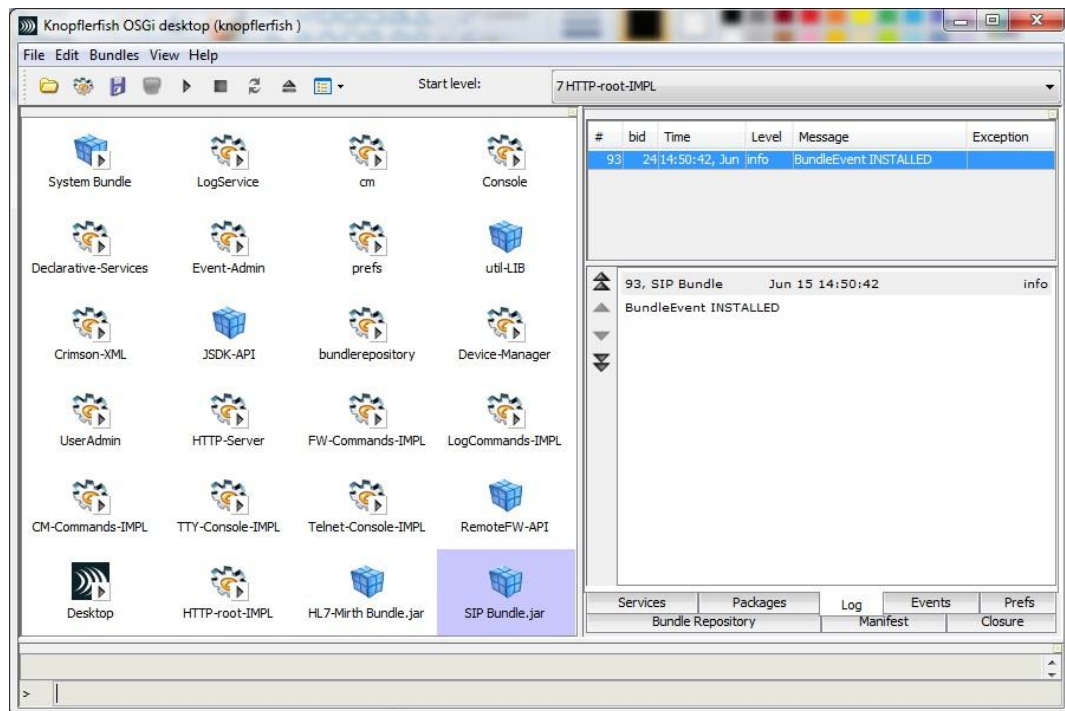


Figura 17. Bundles OSGi de Servidores

5.5 APIs Desarrollados

Se han desarrollado una serie de APIs, para la extensión de las características SIP, en equipos móviles. Este API, aporta el soporte a movilidad y el uso de credenciales en la autenticación. También permite al dispositivo el envío y recepción de mensajes.

Se ha implementado un APIs para el acceso a sensores que dispongan de tecnologías Bluetooth y Socket. Este acceso está condicionada al tipo de conectividad, ya sea síncrona o asíncrona, con lo cual se liberan recursos del sistema. Se soporta protocolos como TCP/IP sobre Bluetooth, con lo que se podría tener una interacción entre sensores homogénea basada en tecnologías de Internet.

5.5.1 API SIP J2ME

```
package cd.interfaz;

/**
 *
 * @author Augusto
 */
public interface InterfazSIP {

    /**
     * Registra un usuario sip con las credenciales y el URI proporcionado
     *
     * @param username SIP-URI para autenticar ej. sip:upmovil@dit.upm.es
     * @param password contraseña para registrarse
     * @return boolean true si el registro fue satisfactorio, false si no lo fue
     * @throws IllegalArgumentException Si el nombre es null o vacio
     */
    public boolean registrarSIP(String username, String password, String realm);

    /**
     * Registra un usuario sip con las credenciales y el URI proporcionado
     *
     * @param mensaje Mensaje a Enviar
     * @param parametros algun parametro especial de comunicacion, importancia etc.
     * @param idComunicaion id de comunicacion para mantener estado/id de llamada
     * @return String Con el codigo de Error en caso de falla
     * @throws IllegalArgumentException Si el nombre es null o vacio
     */
    public String enviarMensaje(String mensaje, String parametros, int idComunicacion) throws
    IllegalArgumentException;;

}
```

5.5.2 API Otras Tecnologías

```
/**
 * Interfaz J2ME Para el Acceso hacia Distintos Sensores
 * @author UPM
 * @version 0.1
 */
public interface Interfaz {

    /**
     * Se conecta con un dispositivo, ya sea para obtener su estado, inicializarla etc...
     * @param tipo Tipo de tecnología del sensor e.g: Bluetooth, Socket, Local
     * @param direccion Direccion logica del sensor e.g: socket://localhost:30,
    btgoep://localhost:1452145
     * @param metodo Metodos que acepta la capacidad, e.g. Connect (Bluetooth); /iniciarservicio
    (REST); init() (INTERNA)
     * @return String Con el estado de la conexion, si fue satisfactoria o no e.g NULL, OK, ERROR etc.
     */
    public DataAccesoM connect(String tipo, String direccion, String metodo, String param1,String
    param2);

}
```

```

    /**
     * Se desconecta de un dispositivos
     * @param tipo Tipo de tecnología del sensor e.g: Bluetooth, Socket, Rest, Local
     * @param direccion Direccion logica del sensor e.g: socket://localhost:30,
    btgoep://localhost:1452145
     * @param metodo Metodos que acepta el sensor, e.g. Disconnect (Bluetooth); close() (INTERNAL)
     * @return String Con el estado de la conexión, si fue satisfactoria o no e.g NULL, OK, ERROR etc.
     *
     */
    public DataAccesoM disconnect(String tipo, String direccion, String metodo, String param1,String
    param2);

    /**
     * Invoca un método de una dispositivo síncrono
     * @param tipo Tipo de tecnología del sensor e.g: Bluetooth, Socket, Local
     * @param direccion Direccion logica del sensor e.g: socket://localhost:30,
    btgoep://localhost:1452145
     * @param metodo Metodos que acepta el sensor, e.g. PUT,GET (Bluetooth);
     * @return String Con el resultado de la invocación
     *
     */
    public DataAccesoM invokeSync(String tipo,String direccion, String metodo,String param1,String
    param2);

    /**
     * Invoca un método de un dispositivo asíncrona
     * @param tipo Tipo de tecnología del sensor e.g: Bluetooth, Socket, Local
     * @param direccion Direccion logica del sensor e.g: socket://localhost:30,
    btgoep://localhost:1452145
     * @param ID Identificador de llamada del armonizador, con el cual se podrá recuperar el ID del
    dato que se solicitó asíncronamente
     * @param metodo Metodos que acepta el sensor, e.g. PUT,GET (Bluetooth);
     * @return String Con el resultado de la invocación
     *
     */
    public DataAccesoM invokeASync(String tipo,String direccion, String metodo, String ID, String
    param1,String param2);

}

```

5.6 Conclusiones

En este capítulo se exponen los desarrollos que se han hecho para validar la arquitectura descrita en este trabajo de Fin de Master. Algunos de las funcionalidades diseñadas están pendientes de despliegue en un entorno real hospitalario. Con esto se pretende estudiar el comportamiento de la arquitectura frente a situaciones reales, así como la interacción con el personal. Uno de los aspectos interesantes es que se podrán obtener datos relativos a los sensores desplegados, de manera que se adaptará el modelo de datos previamente propuesto en capítulos anteriores.

Los desarrollos expuestos aquí están sustentados en los proyectos CARDEA y CARDINEA, y serán mejorados según la disponibilidad de recursos y los resultados que se esperan obtener.

6 Arquitecturas de Red Hospitalarias para Situaciones de Emergencia

Las arquitecturas de red hospitalarias además de soportar los servicios propios de un hospital general, deben ser adaptables a situaciones de emergencia. En este capítulo describe la posible aplicación de conceptos y la arquitectura descrita en los capítulos anteriores a un tipo de escenarios hospitalarios donde se contemplan situaciones de emergencia y se aborda su solución mediante el uso de nuevas tecnologías.

Un análisis sobre las arquitecturas de red hospitalarias para situaciones de emergencias debe comprender la inclusión de múltiples eventos, funcionalidades, requisitos y elementos de red dinámicos. Está claro que este tipo de arquitectura debe tener la movilidad como una de sus principales características, ya que por definición según la RAE *“una emergencia es una situación de peligro o desastre que requiere una acción inmediata”*, por lo que en estos casos no hay tiempo para adaptar todos los elementos y dispositivos relacionados con una infraestructura de red fija. Este tipo de eventos causa que servicios tan básicos como la búsqueda o envío de mensajes puedan quedar inhabilitados. Este capítulo abordará los temas de despliegue de servicios dinámicos desde un punto de vista conceptual.

La inclusión de movilidad en ambientes hospitalarios ha sido probada como un concepto coherente y acertado[27][28]. Característica que ya ha sido planteada previamente en este documento, por lo que la utilización de SIP y sus características nativas, es planteada como una de las maneras de superar las desventajas de una infraestructura de red fija. Pero existe un inconveniente ya que la utilización de SIP puro, está relacionado con el despliegue de servidores fijos, por lo que al final, este tipo de soluciones están limitadas a la existencia previa de este tipo de nodos dentro de la red.

6.1 Arquitecturas P4P

Adicionalmente a las infraestructuras estáticas de cliente/servidor existen otros tipos que se enfocan en soportar la inclusión de servicios mediante topologías dinámicas P2P.

Los orígenes de P2P son diversos, pero el aspecto que más ayudó a popularizar este tipo de tecnologías fue el intercambio de archivos multimedia. De la misma manera se han desarrollado múltiples aplicaciones y protocolos para este modelo de red, además de aplicaciones comerciales. Adicionalmente las arquitecturas P2P han sido explotadas en los últimos años como una forma eficaz de distribuir contenido dentro de Internet y han sido un modelo de éxito que se está aplicando otros campos no pensados originalmente.

Existen recientes estudios[30][29] que sugieren el uso de las arquitecturas P2P, para su aplicación en entornos médicos y en situaciones de emergencias[31] que requieran una rápida adaptación del sistema. Sin embargo para lograr los niveles de calidad de servicio, disponibilidad y fiabilidad necesarias por alguno de los servicios médicos catalogados como críticos, es necesaria una cooperación con la propia infraestructura de red hospitalaria que ha sido desplegada previamente.

Una arquitectura P4P es una evolución de una arquitectura P2P, y como resultado comparte sus características principales de descentralización y redundancia. Fue ideada para resolver el problema de acceso a recursos desde las redes P2P. La idea principal radica en la participación de los Proveedores de Internet en la provisión de datos que ayuden a encontrar los recursos, y conocer cuáles son los nodos más cercanos que los comparten. La aportación principal de P4P radica en un enfoque más colaborativo de la red, en donde proveedor con una infraestructura fija y operativa, puede utilizar todos los recursos disponibles, para ayudar a clientes que forman parte de una red dinámica a encontrar la información que buscan. En resumen, la arquitectura P4P provee, por medio de funcionalidades específicas, información que pueda ser útil a los nodos P2P, con el fin de optimizar el rendimiento general de la red.

Los sistemas basados en redes P4P, se basan en una cooperación con los sistemas e infraestructuras fijas ya existentes, de manera que los nodos pueden obtener toda la información que necesitan sobre la topología de red. La información obtenida es útil para realizar las comunicaciones a nivel IP de manera optimizada. Sin embargo las arquitecturas P4P como tal están enfocadas a nivel de red, es decir cómo se encaminan los paquetes a través de las redes, pero no abordan los temas sobre interacción de interacción, búsqueda, despliegue y optimización a nivel de los servicios que son desplegados sobre la plataforma.

6.2 Evolución de P4P como Arquitectura para Situaciones de Emergencias

Un sistema P4P siguiendo un enfoque orientado a servicios, puede traducirse como una cooperación entre una infraestructura de red fija, y una dinámica, en donde la fija puede proveer un soporte a recursos que no estén disponibles en una dinámica. Este soporte puede ser dividido en: búsqueda, acceso y optimización.

Estos conceptos puede ser llevado a situaciones de emergencia hospitalaria, en donde sea necesario desplegar una red dinámica por diversas razones, entre las cuales están:

- No existencia previa de infraestructura de comunicaciones, o destrucción de la existente.
- Lejanía de los lugares en donde deben ser desplegados los servicios
- Colapso de comunicaciones debido al alto flujo de tráfico
- Multiplicidad de sensores médicos
- Elementos móviles que requieran altas prestaciones de la red.

Cada uno de estos escenarios plantea unos requisitos que han de ser analizados a nivel de red, pero existen otros escenarios más globales que pueden ayudar a entender la necesidad de una arquitectura P4P.

Para hacer una primera aproximación de una arquitectura P4P hospitalaria que soporte situaciones de emergencias se ha definido el escenario descrito por la figura 18. En este caso específico se han ideado sub-escenarios que representan dominios de comunicación, junto con las características que los representan.

El dominio número 1 representa una infraestructura de red hospitalaria estática en donde todos los recursos disponibles han sido planificados según la cantidad del tráfico y de nodos que existirán. A esto se le añaden los servicios que son proveídos por distintas instancias de red. Está claro que en una arquitectura P4P este dominio será categorizado como de altas prestaciones.

El dominio 2 representa una infraestructura fija, similar a la primera, pero de no tantas prestaciones, ya que es un hospital de campaña temporal.

El dominio 3 describe una situación de emergencia en donde un elemento móvil ya sea ambulancia u otro. Se han desplegado una serie de pequeños centros de atención que a su vez necesitan conectividad con los otros dominios de comunicación. En este caso una infraestructura de red dinámica ha sido desplegada, la cual no tiene altas exigencias.

Finalmente el dominio 4, presenta las mismas características del dominio 3, en donde una red dinámica ha sido desplegada, sin embargo altas prestaciones son requeridas para proveer servicios críticos hospitalarios.

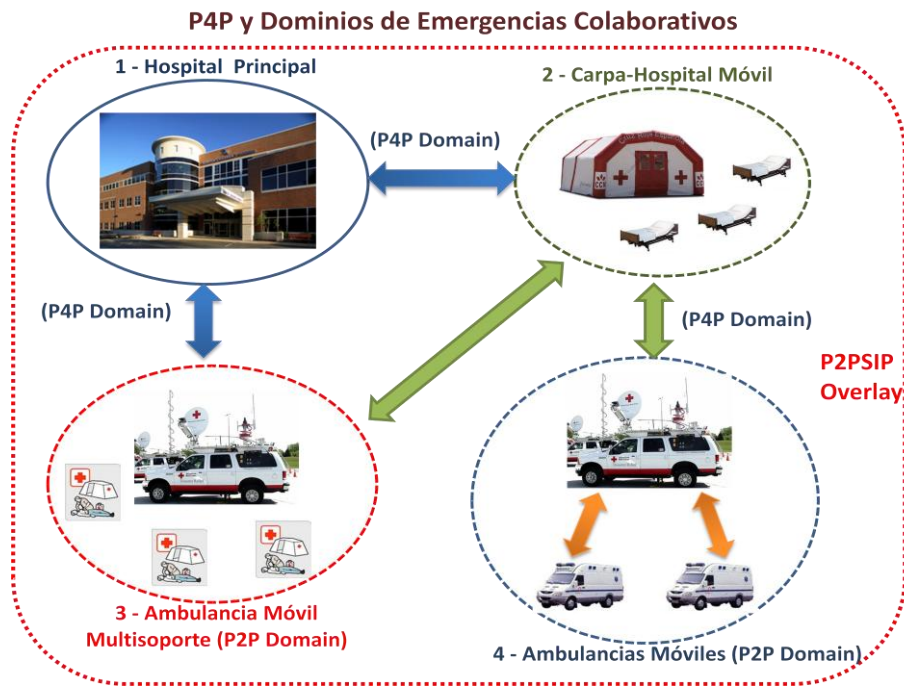


Figura 18. Arquitectura P4P para Situaciones de Emergencias

En los dominios 3 y 4 se nota la necesidad de una infraestructura de red dinámica que debe apoyarse sobre otra estática, ya que las propias características de esta, como lo podrían ser ancho de banda, servicios ofrecidos, o fiabilidad no pueden ser suplidas. Según estas características una arquitectura P4P a nivel de red y de servicios podría ser de gran ayuda, ya que los propios dispositivos de red, y los diversos servicios provistos serían conscientes de las capacidades de la red, topología desplegada en este momento.

Una de las consideraciones para el despliegue de una arquitectura colaborativa de red, orientada a las situaciones de emergencia es la posibilidad de soportar servicios que aparecen, desaparecen, se crean y se consumen, pero que a su vez aportan información de sus características y restricciones a la red. Para ello la utilización de protocolos como P2PSIP[32], y sus paradigmas de publicación de servicios en redes overlay, permite que haya una descubrimiento de los servicios que han sido desplegados en una red dinámica. En una arquitectura hospitalaria existen elementos de red que no proveen servicios como tal, sino información para sustentar otros servicios, como lo son sensores biomédicos, equipos de comunicación y monitoreo etc. En este caso la incorporación de estos elementos a la red mediante P2PSIP extendería las posibilidades de una futura composición de servicios hospitalarios, mediante una homogenización en los protocolos y el método de acceso a estos elementos. Claro está que para ello sería necesaria la existencia de algún middleware de adaptación que

podiera abstraer estas funcionalidades a las capas superiores que gestionan estos nuevos elementos de red.

Aunque en este capítulo se ha abordado muy superficialmente los mecanismos necesarios para una interacción entre servicios, redes, infraestructuras dinámicas y estáticas en ambientes hospitalarios y de emergencia, se ha identificado un modelo de red basado en P4P que puede ser útil para su aplicación en casos de emergencia, además de la utilización de P2PSIP como mecanismo de publicación y búsqueda. Con esto se cumplen los objetivos propuestos en el punto 1.3. Parte de las deducciones hechas en este capítulo han sido resultado del análisis y unificación de conceptos previamente planteados en el proyecto CARDINEA, y CARDINEA junto con el enfoque orientado a servicios del proyecto mIO!.

7 Conclusiones y Trabajos Futuros

La definición de una arquitectura detallada es una característica necesaria en cualquier sistema de comunicación hospitalario. La arquitectura permite definir por medio de módulos, todos los requisitos para que un servicio médico pueda ser ofrecido bajo un canal de comunicación, y unas condiciones optimizadas.

En este trabajo de Fin de Master se ha propuesto un desarrollo de una arquitectura orientada a servicios hospitalarios. Como primera parte se ha hecho una revisión del estado del arte de las TIC y las distintas propuestas para el despliegue de servicios hospitalarios, además de un análisis de sus requisitos.

Se han analizado casos de uso específicos en los cuales se identifica las funcionalidades que son necesarias para una adaptación de dispositivos a un Hospital del Futuro. Adicionalmente se ha propuesto una arquitectura detallada de todos los elementos implicados, una topología de referencia, y unos modelos de datos que permiten clasificar la importancia de los datos médicos y los requisitos que imponen hacia la red.

Como validación de los conceptos expuestos y como parte de los trabajos realizados en distintos proyectos se describen los desarrollos y prototipos realizados siguiendo las pautas dictadas por la propuesta inicial. Estos prototipos han demostrado que son viables los conceptos y funcionalidades descritos en este trabajo. Para obtener más información sobre estos prototipos será necesario analizar los datos provenientes del sistema completo que será implementado en un Hospital español.

Uno de los aspectos más importantes de la arquitectura, es que se han definido mecanismos de interoperabilidad con sistemas existentes en hospitales. Se han tomado en cuenta estándares que permiten una fácil integración entre los distintos elementos o funcionalidades futuras que puedan ser adicionadas tanto a los nodos como a los servidores.

Se han considerado aspectos de gestión de la red, y su interacción con elementos de alto nivel. Con esto, se ha planteado una adaptación de los enlaces y sesiones de comunicaciones a los requisitos dictados por sensores, dispositivos médicos o situaciones específicas que pueden incluir acciones del personal médico. El soporte de movilidad mediante la utilización de SIP asegura la interacción entre distintos elementos de monitorización y alarma, además de su compatibilidad con tecnologías

de Internet. En la gestión de red se ha aportado una posible solución para el soporte de picos de tráfico que puedan afectar el funcionamiento e integridad de procesos críticos.

Se han desarrollado funcionalidades que permitan la futura inclusión de sensores con distintas tecnologías de acceso, y una primera aproximación conceptual sobre cómo integrarlos dentro de la arquitectura IP de la red hospitalaria. Aspectos como la monitorización del personal y pacientes ha sido tomada en cuenta con la aplicación de tecnologías como RFID. Las funcionalidades propuestas están basadas en estándares abiertos, por lo que la arquitectura se clasifica como abierta para futuras adaptaciones según el entorno.

La propuesta de esta arquitectura hospitalaria y su análisis, plantea múltiples líneas de investigación futuras, que incluyen desarrollos a varios niveles, desde el nivel de red, hasta el nivel de servicios y aplicación. Además los conceptos propuestos en el capítulo 6, sobre colaboración entre redes dinámicas y redes fijas aportan guías sobre cómo adaptar las necesidades de las redes hospitalarias en casos de emergencia.

Según el trabajo de investigación realizado, se perciben múltiples líneas de investigación de cara a una arquitectura hospitalaria integral, que aborde una completa gestión de servicios y una adaptabilidad al contexto y las diversas situaciones que pudieran ocurrir.

7.1 Trabajos Futuros

A continuación se describen posibles líneas de investigación y los problemas que pueden ser abordados:

- Extensión hacia una arquitectura completamente abierta: aunque la arquitectura propuesta está diseñada para la inclusión de distintos tipos de servicios está claro que la inclusión de ciertas funcionalidades que no sean compatibles tecnológicamente con OSGi presentan un reto que será abordado. Además de ello la compatibilidad de servicios, sistemas o nodos que sean de distintos proveedores está limitado a la complejidad o importancia de las funcionalidades ofrecidas, además de las propias limitaciones de software o hardware que se puedan tener.
- Colaboración entre infraestructuras de red: como fue explicado en el capítulo 6, un problema de las arquitecturas hospitalarias en casos de emergencia es el dinamismo con que deben ser desplegadas, así como las necesidades de autoconfiguración, y adaptación de los nodos que formen parte de la red. Un problema existente consiste en cómo construir un sistema dinámico que soporte múltiples nodos, y que a su vez sea consciente de los servicios que son desplegados en la red. Luego de esto, cómo este sistema o infraestructura

dinámica pueda interactuar con una estática, a la vez que se sirve de funcionalidades que no pueden ser soportadas por ella misma. Esta colaboración entre infraestructuras deberá estar sustentada en alguna métrica o modelo que la sustente, además de incluir situaciones variables y de emergencia, que puedan influir en la valoración de los nuevos servicios que han sido creados. Una futura línea de trabajo se centrará en definir de manera formal estas relaciones.

- Escalabilidad de servicios hospitalarios con voz, video, mensajes. Los sistemas actuales hospitalarios soportan estos tipos de servicios, y la arquitectura propuesta los toma en cuenta mediante la inclusión del protocolo SIP. Pero existen problemas de escalabilidad sobre todo en situaciones de emergencias, en donde grandes volúmenes de tráfico son generados. Trabajos futuros se centrará en la identificación de situaciones específicas que requieran altas tasas de ancho de banda, fiabilidad y baja latencia.
- Combinación de servicios: los paradigmas actuales de Internet han fortalecido la combinación de servicios mediante tecnologías web. Este tipo de enfoques podría ser aplicado a ambientes médicos. La combinación de servicios médicos es una línea futura de trabajo ya que permite adaptar tecnologías de Internet, a las restricciones impuestas en ambientes de trabajo.
- Integración de usuarios que crean, y consumen servicios. La arquitectura propuesta abre la posibilidad para una interacción entre el personal médico, y pacientes con los servicios disponibles en la plataforma. En este contexto se podrán analizar, en el futuro, los mecanismos de creación de servicios, de manera que se le aporte una flexibilidad en el sistema, en donde los propios usuarios finales puedan elegir la manera en que los distintos elementos (sensores, equipos etc.) interactúan entre sí.
- Modelo de integración de sensores. Una futura integración de un modelo de sensores dentro de la arquitectura, supondría una mejora a la arquitectura global ya que mediante la estandarización IP, se podrían gestionar a alto nivel todos los datos provenientes de sensores.

Publicaciones Relacionadas con este Trabajo

- *Augusto Morales Domínguez, Tomas Robles Valladares, Ramón Alcarria, David Alonso, Silvia Platas Bricio "Communication Architecture for Smart Services in Hospital Environments" International Workshop of Ambient Assisted Living 2010.*

Este artículo propone una arquitectura de comunicación para hospitales. Se definen los modelos de datos, nodos, y funcionalidades de que deben ser implementadas en servidores. Se utilizan las tecnologías SIP y OSGI como base.

- *Sergio González-Miranda, Tomás Robles, Augusto Morales, Ramón Alcarria "Converged Service Provision in Multidomain Environments for the Future Internet". ICIN 2009.*

Se exponen conceptos de convergencia de servicios en múltiples ambientes de Internet. La provisión de servicios mediante IMS es abordada así como servicios de IPTV y presencia.

- *Augusto Morales Domínguez, Tomás Robles, Ramon Alcarria, Sergio González-Miranda "Arquitectura para Provisión de Servicios Ubicuos en redes IMS P2P" IX Jornadas de Ingeniería Telemática (JITEL 2010)*

Se expone una plataforma de despliegue de servicios ubicuos. Se justifica la colaboración entre redes P2P e infraestructuras dinámicas. También es explicada una futura interoperabilidad entre IMS-P2P a nivel de servicio

- *Ramón Alcarria, Tomás Robles, Augusto Morales Domínguez "Plataforma de composición, provisión y consumo de servicios para el nuevo universo inteligente" IX Jornadas de Ingeniería Telemática (JITEL 2010)*

Este artículo trata de la composición de servicios y la necesidad de incluir a los usuarios como elementos creadores. La inclusión de usuarios que despliegan, y consumen servicios mediante lenguajes estandarizados, así como los problemas de búsqueda de componentes.

- *David Alonso, Tomás Robles, Augusto Morales, Ramón Alcarria "Desarrollo de una Arquitectura de Comunicaciones para Transmisión de Señales Médicas en Entorno Hospitalario" IX Jornadas de Ingeniería Telemática (JITEL 2010)*

Este artículo se enfoca en la parte de desarrollo de una arquitectura hospitalaria, y presenta las características del prototipo que se está implementando en el proyecto CARDINEA.

Bibliografía

- [1] Koop, C.E. ; Mosher, R. ; Kun, L. ; Geiling, J. ; Grigg, E. ; Long, S. ; Macedonia, C. ; Merrell, R. ; Satava, R. ; Rosen, J. ; C. Everett Koop Inst., Dartmouth, MA. "IEEE Engineering in Medicine and Biology Magazine" November 2008
- [2] Daniels Coll. of Bus., Univ. of Denver, Denver, CO, USA "A Multi-Period Study of the Evolution of Collaboration in Telemedicine" 43rd Hawaii International Conference on System Sciences (HICSS), 2010
- [3] Komiya, R. "A proposal for telemedicine reference model for future standardization" Proceedings of 7th International Workshop on Enterprise networking and Computing in Healthcare Industry, 2005. HEALTHCOM 2005.
- [4] Choi, Y.B.; Krause, J.S.; Hyewon Seo; Capitan, K.E.; Kyusuk Chung; "Telemedicine in the USA: standardization through information management and technical applications" Communications Magazine, IEEE 2006
- [5] D. Ziadlou, A. Eslami, H.R. Hassani "Telecommunication methods for implementation of telemedicine systems In crisis! Third International Conference on Broadband Communications, Information Technology & Biomedical Applications, 2008 .
- [6] HAPI <http://hl7api.sourceforge.net>
- [7] A. Sadat, G. Sorwar, and M. U. Chowdhury "Session Initiation Protocol (SIP) based Event Notification System Architecture for Telemedicine Applications" 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse, 2006
- [8] Nam-Ho Kim, Yi-Seok Jeong, Seung-Jae Song, Dong-Ryeol Shin. "Middleware Interoperability based Mobile Healthcare System" The 9th International Conference on Advanced Communication Technology, 2007.
- [9] Wen-Wei Lin, Yu Hsiang Sheng. "Using OSGi UPnP and Zigbee to provide a wireless ubiquitous home healthcare environment" The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, 2008. UBICOMM '08.
- [10] Saúl Navarro, Silvia Platas, Ramón Alcarria "CARDEA. Una plataforma OSGi para entorno Hospitalarios" VIII Jornadas de ingeniería Telemática, JITEL 2009.
- [11] Varshney, U.: "Improving Wireless Health Monitoring Using Incentive-Based Router Cooperation". Computer, pp. 56-62, May 2008
- [12] Byungil Lee and Howon Kim. "Ubiquitous RFID based Medical Application and the Security Architecture in Smart Hospitals" International Conference on Convergence Information Technology, 2007
- [13] Hossein Fariborzi, Mahmoud Moghavvemi "Architecture of a Wireless Sensor Network for Vital Signs Transmission in Hospital Setting" International Conference on Convergence Information Technology, 2007
- [14] Haiyong Xie "Explicit Communications for Cooperative Internet Traffic Control" PhD Thesis. Yale University 2008

- [15] Evolved Packet Core (EPC). 3GPP Website.
<http://www.3gpp.org/ftp/Specs/html-info/36-series.htm>
- [16] Ongenaes, F. ; Strobbe, M. ; Hollez, J. ; De Jans, G. ; De Turck, F. ; Dhaene, T. ; Demeester, P. ; Verhoeve, P. ; International Conference on Complex, Intelligent and Software Intensive Systems, 2008. CISIS 2008.
- [17] Zhikui Chen ; Zhe Wei ; Yang Liu ; Yong Piao ; International Conference on Industrial and Information Systems, 2009. IIS '09.
- [18] Byungil Lee and Howon Kim. "Ubiquitous RFID based Medical Application and the Security Architecture in Smart Hospitals" International Conference on Convergence Information Technology 2007.
- [19] Mejla, D.A. ; Favela, J. ; Moran, A.L. ; Transactions on Information Technology in Biomedicine, IEEE. Jan 2010
- [20] Jae-Min Lee ; Jong-Hyouk Lee ; Tai-Myoung Chung ; 10th International Conference on Advanced Communication Technology, 2008. ICACT 2008.
- [21] Chung-Min Chen, Hira Agrawal, Munir Cochinwala, David Rosenbluth "Stream Query Processing for Healthcare Bio-sensor Applications" 2004
- [22] Dae-Seok Lee, Sachin Bhardwaj, Esko Alasaarela and Wan-Young Chung "An ECG Analysis on Sensor Node for Reducing Traffic Overload in u-Healthcare with Wireless Sensor Network" 2007
- [23] Proyecto Cardea. Disponible en: <http://cardea.germinus.com/inicio>
- [24] Proyecto Cardinea. Disponible en <http://cardinea.grupogesfor.com/>
- [25] Knoplerfish <http://www.knoplerfish.org/>
- [26] CARDEA: Service platform for monitoring patients and medicines based on SIP-OSGi and RFID technologies in hospital environment, Saúl Navarro, Ramón Alcarria, Juan A. Botía, Silvia Platas and Tomás Robles, OpenHealth Spain, 2009.
- [27] Vladimir Ergovic 1, Stanko Tonkovic, "SIP - HL7 integration evaluation", 48th International Symposium ELMAR-2006, June 2006
- [28] A. Sadat, G. Sorwar, and M. U. Chowdhury " Session Initiation Protocol (SIP) based Event Notification System Architecture for Telemedicine Applications. Proceedings of the 5th IEEE/ACIS International Conference on Computer and Information Science, 2006
- [29] Lican Huang "Semantic P2P Network for healthcare" Conference on INC, IMS and IDC, 2009. NCM '09. Fifth International Joint " November 2009
- [30] Hung Keng Pung ; Tao Gu ; Wenwei Xue ; Palmes, P.P. ; Jian Zhu ; Wen Long Ng ; Chee Weng Tang ; Nguyen Hoang Chung "Context-aware middleware for pervasive elderly homecare" IEEE Journal on Selected Areas in Communications, 2009
- [31] Catarci, T. ; Leoni, M.D. ; Rosa, F.D. ; Mecella, M. ; Poggi, A. ; Dustdar, S. ; Juszczak, L. ; Truong, H.L. ; Vetere, G. ; "The WORKPAD P2P Service-Oriented Infrastructure for Emergency Management". 16th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2007. WETICE 2007
- [32] P2PSIP <http://www.p2psip.org/>
- [33] Hussain, M. ; Afzal, M. ; Ahmad, H.F. ; Khalid, N. ; Ali, A. ; " Healthcare Applications Interoperability through Implementation of HL7 Web Service

Basic Profile” Sixth International Conference on Information Technology: New Generations, 2009. ITNG '09.

- [34] Namli, T. ; Aluc, G. ; Dogac, A. ; “An Interoperability Test Framework for HL7-Based Systems” IEEE Transactions on Information Technology in Biomedicine, Mayo 2009